Kafka: Build, Deploy, & Monitor Your First Real-world Application

THE BASICS: TOPICS, PRODUCERS, & CONSUMERS

Justin Pihony
DEVELOPER SUPPORT MANAGER @ LIGHTBEND

@JustinPihony

Course Overview

Kafka Basics

Configuration

Customization & Production Readiness

DevOps: Monitoring

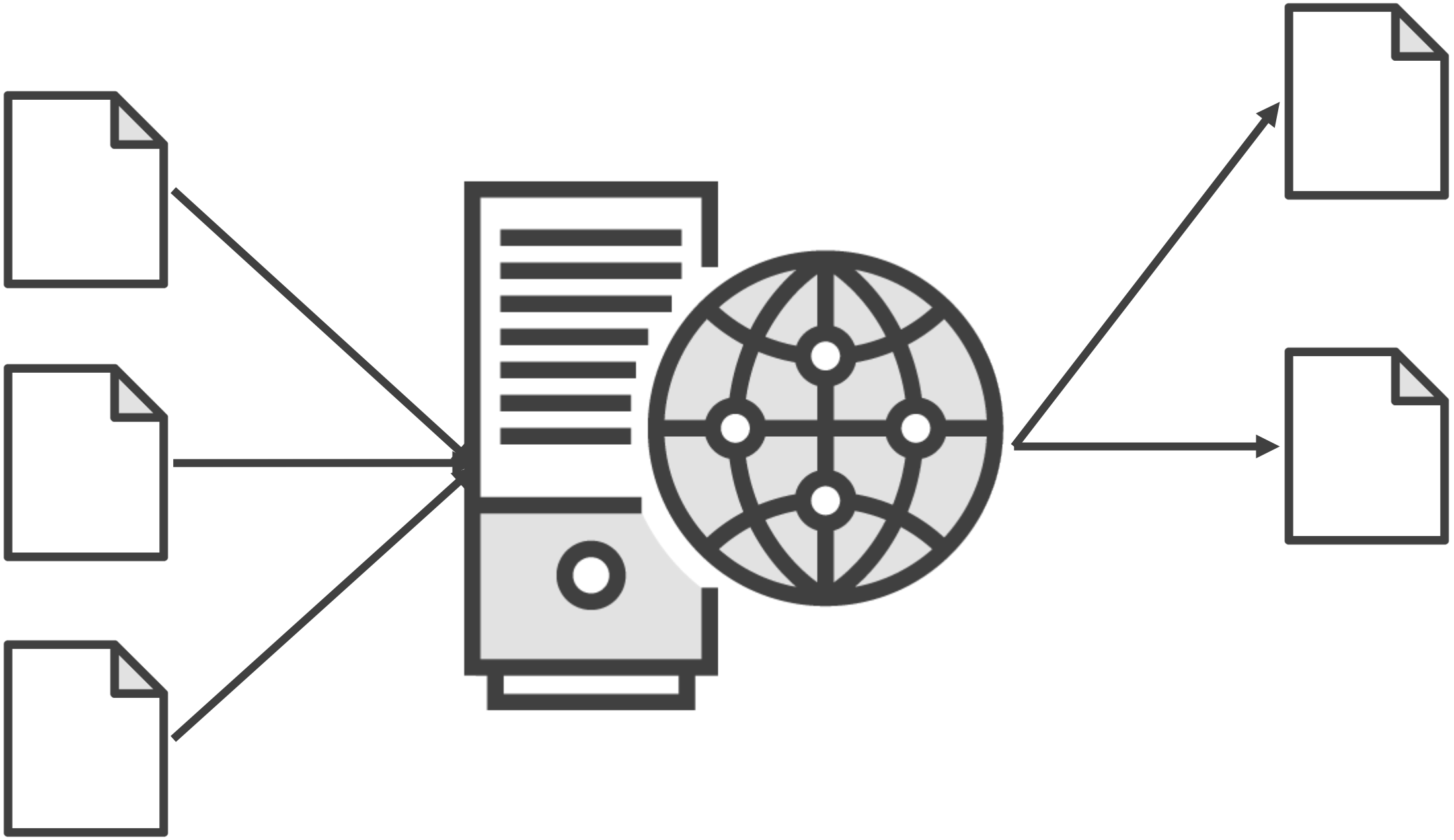
Kafka Basics

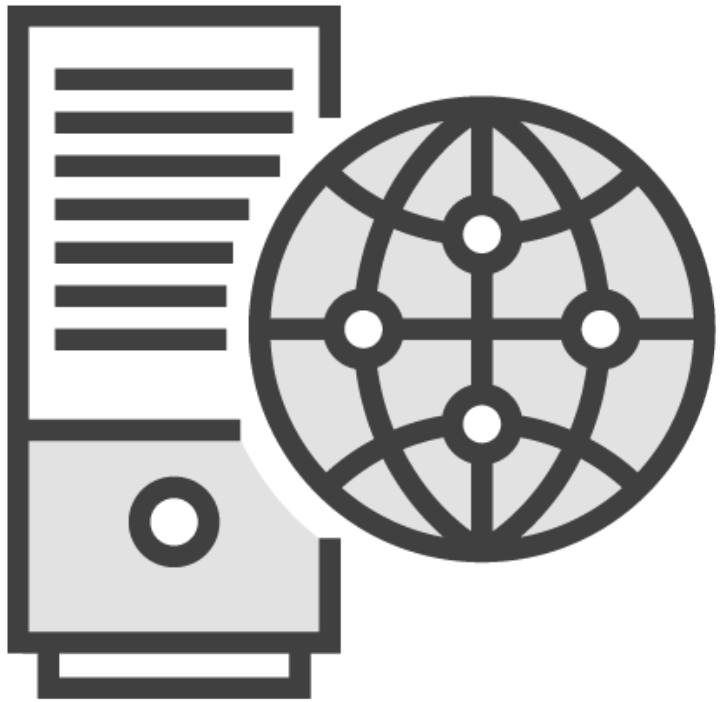
- Brokers
- Zookeeper
- Producers
- Consumers

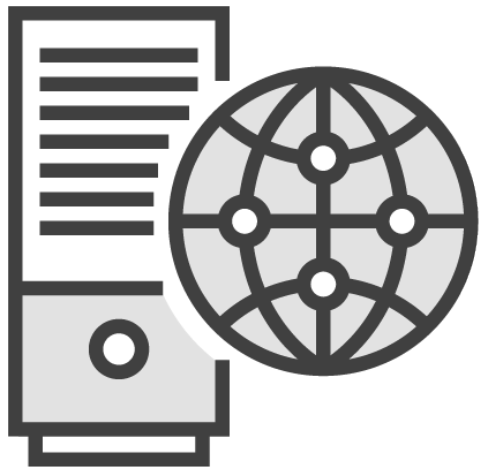
Kafka's Design:
A Perfect Fit For Microservices

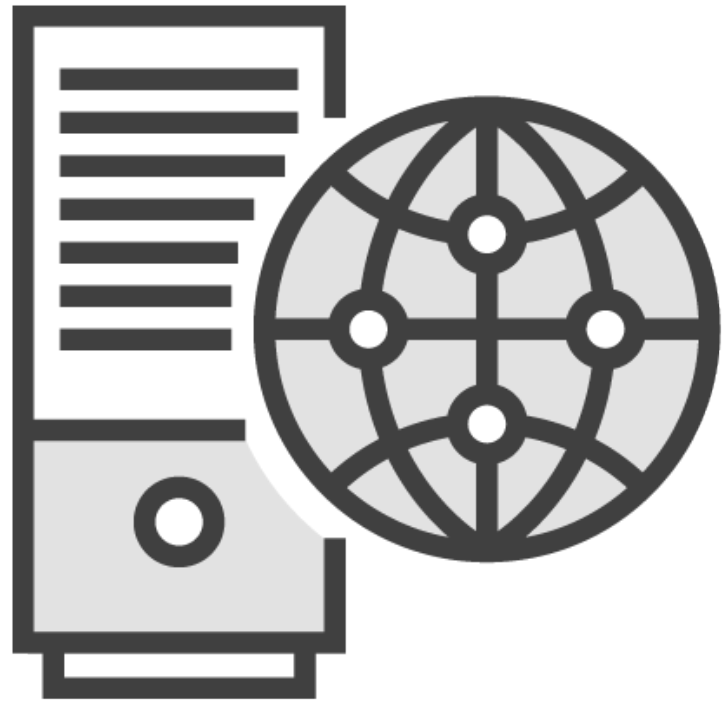
Apache Kafka® is *a distributed streaming platform*

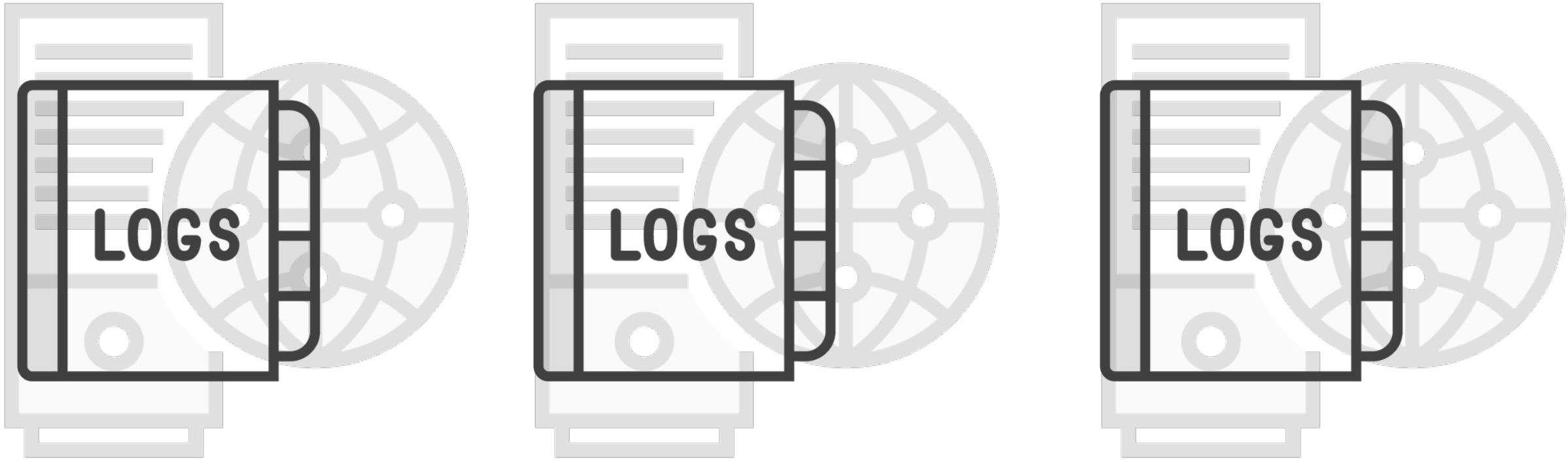
https://kafka.apache.org/intro

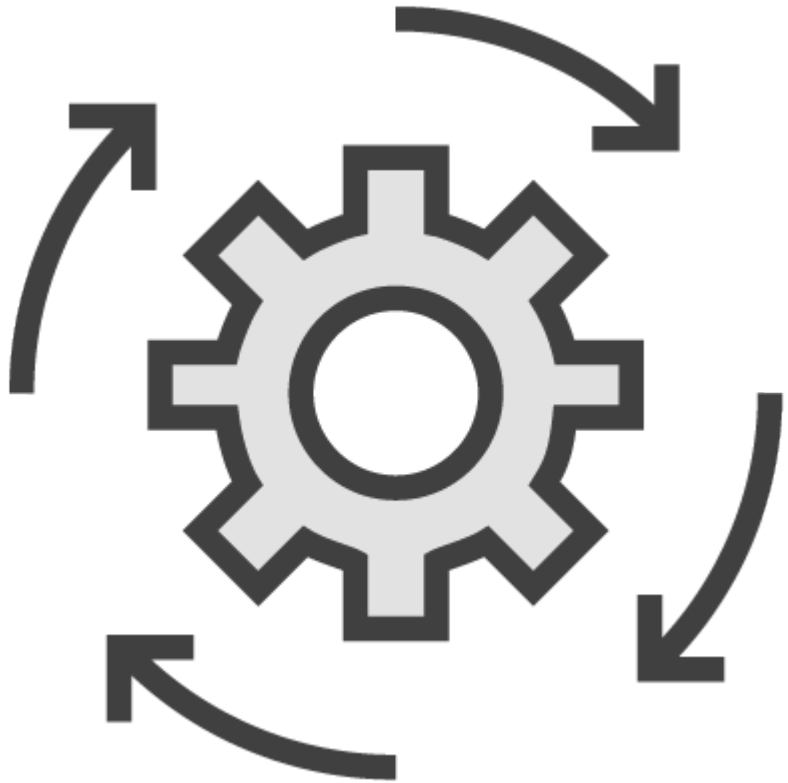




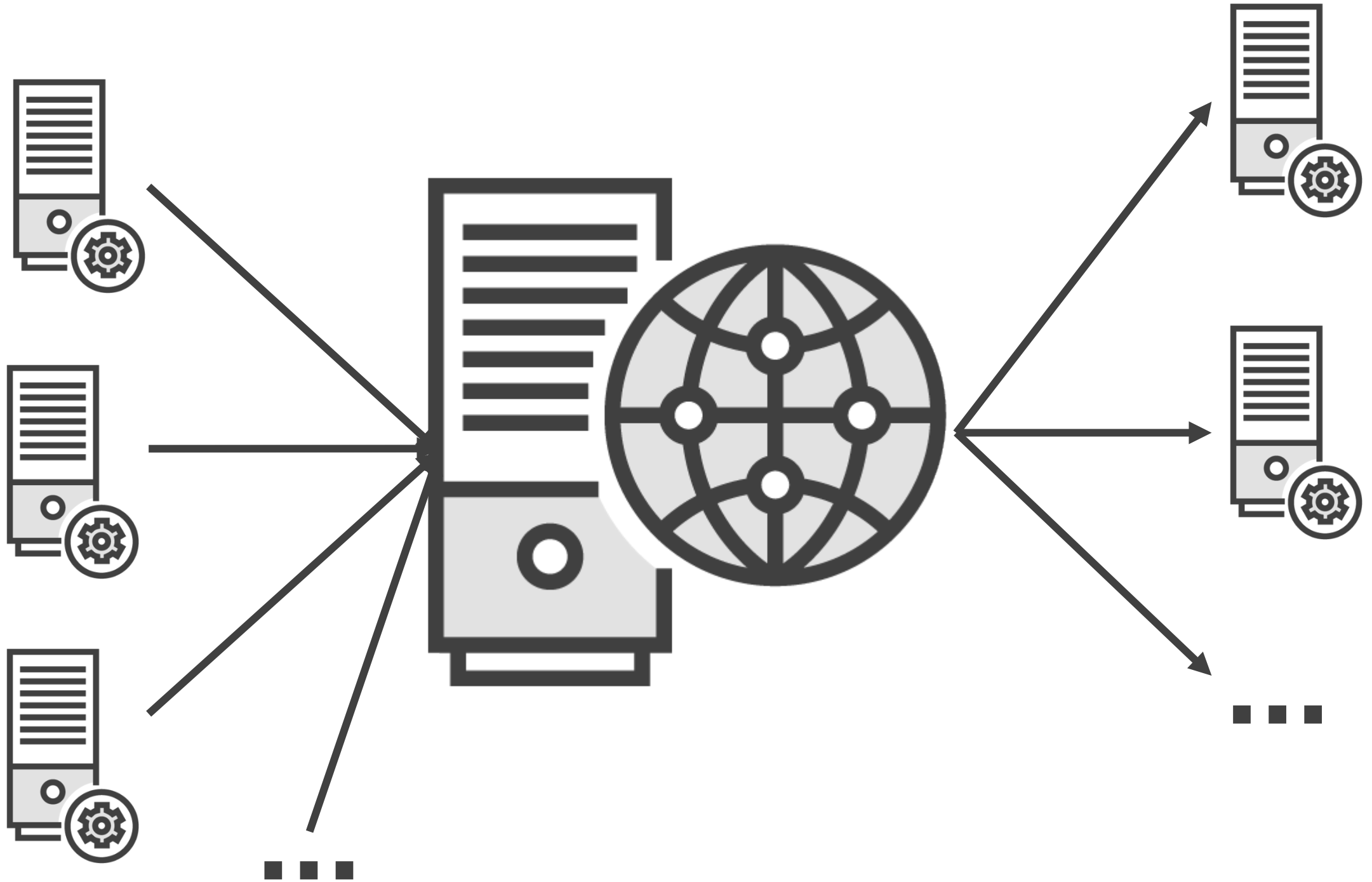


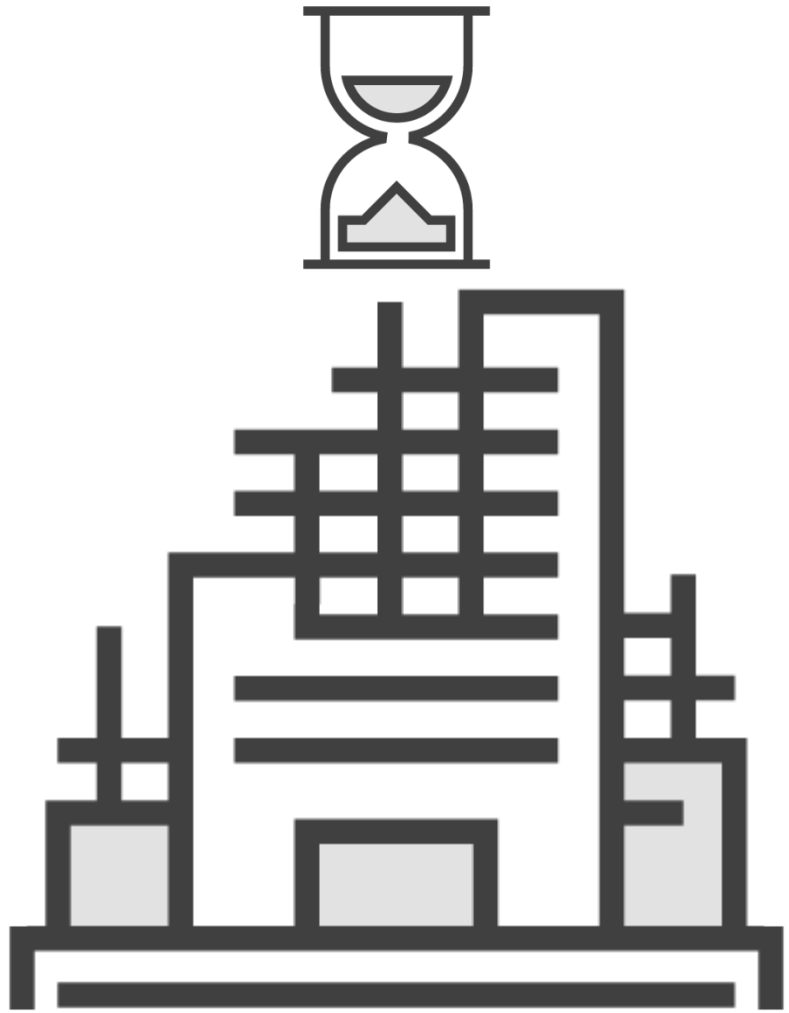


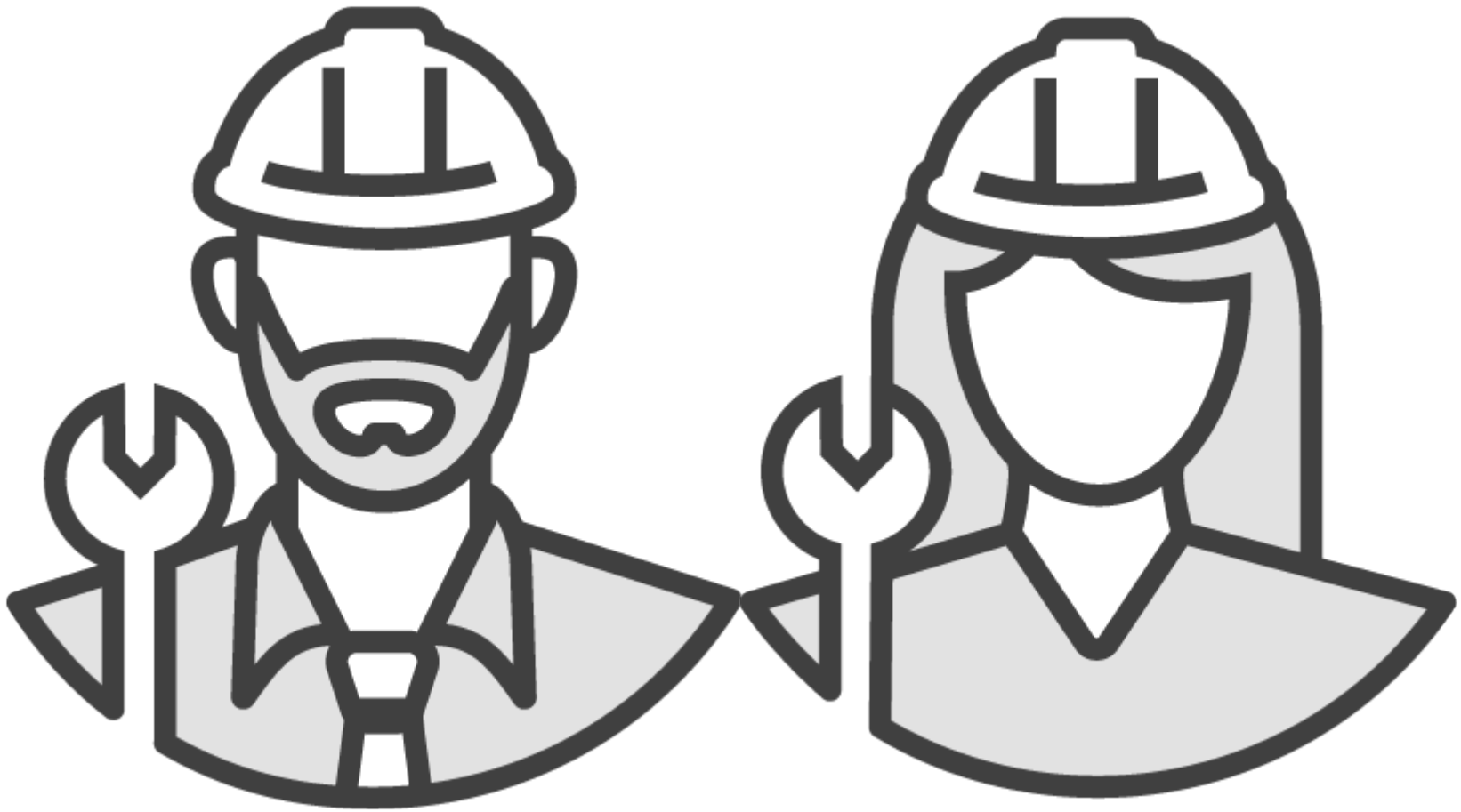

LOGS
LOGS
LOGS

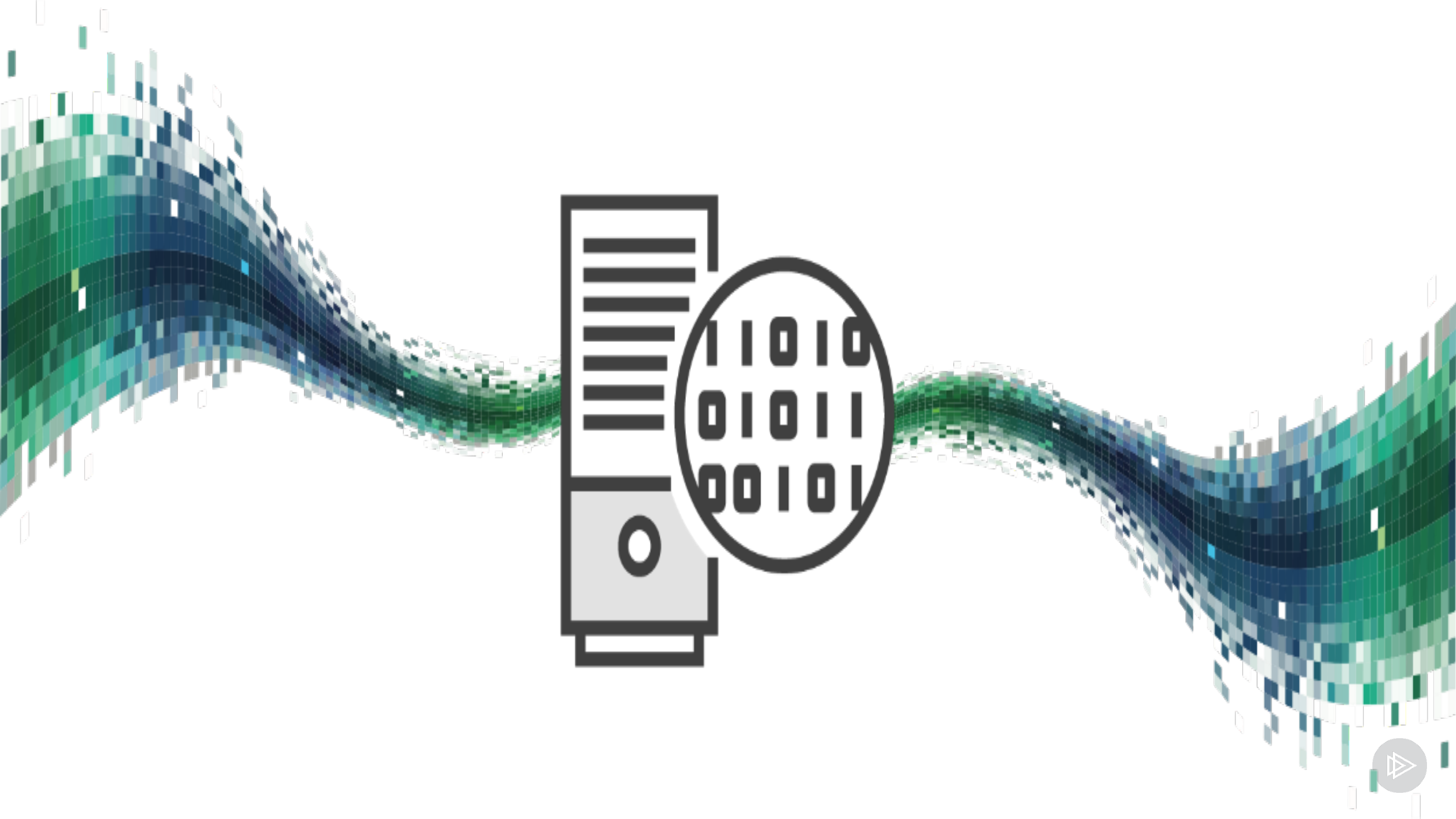


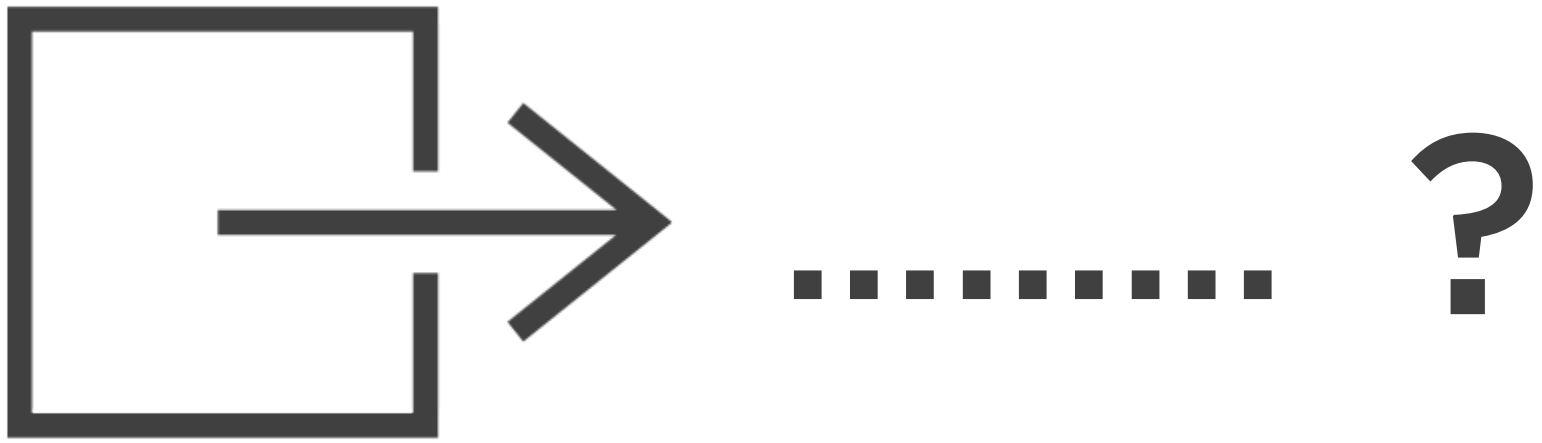
Column 1	Column 2
Data	Data
Data	Data
Data	Data

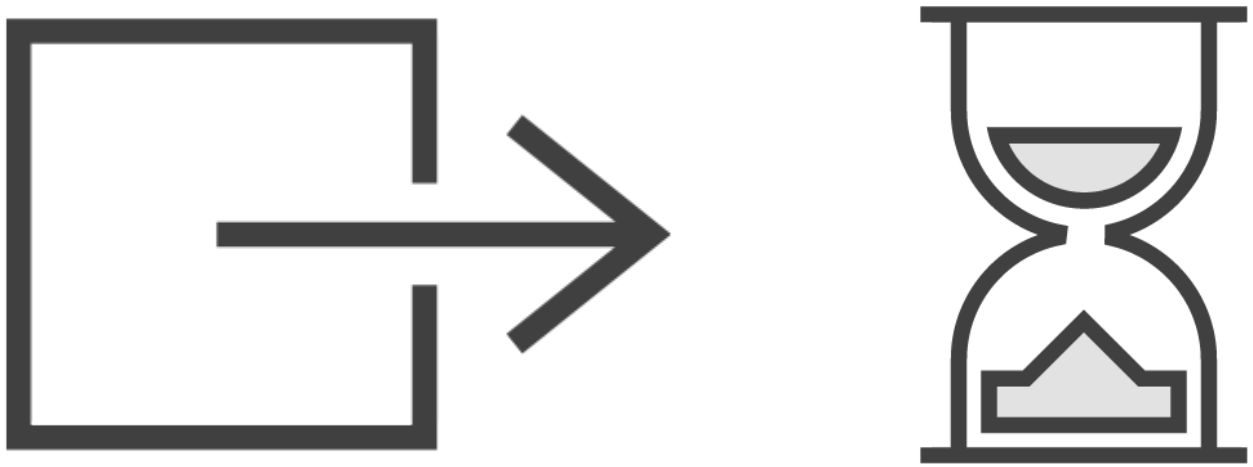


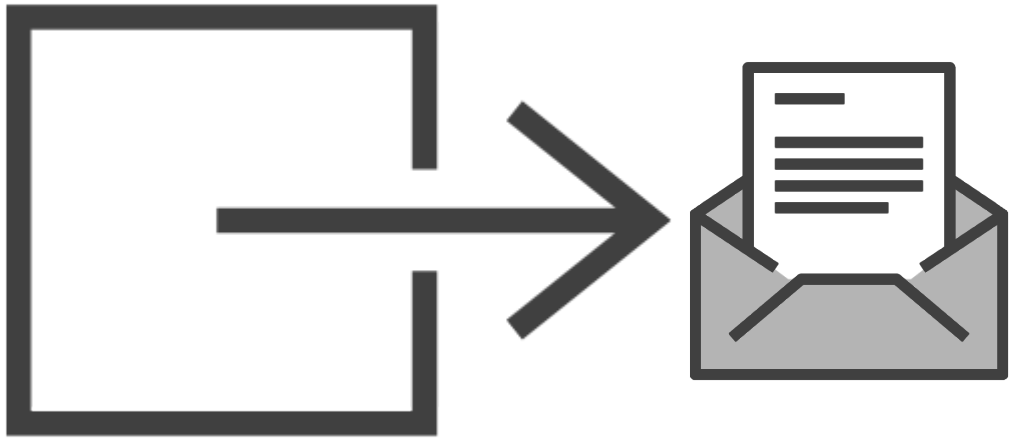




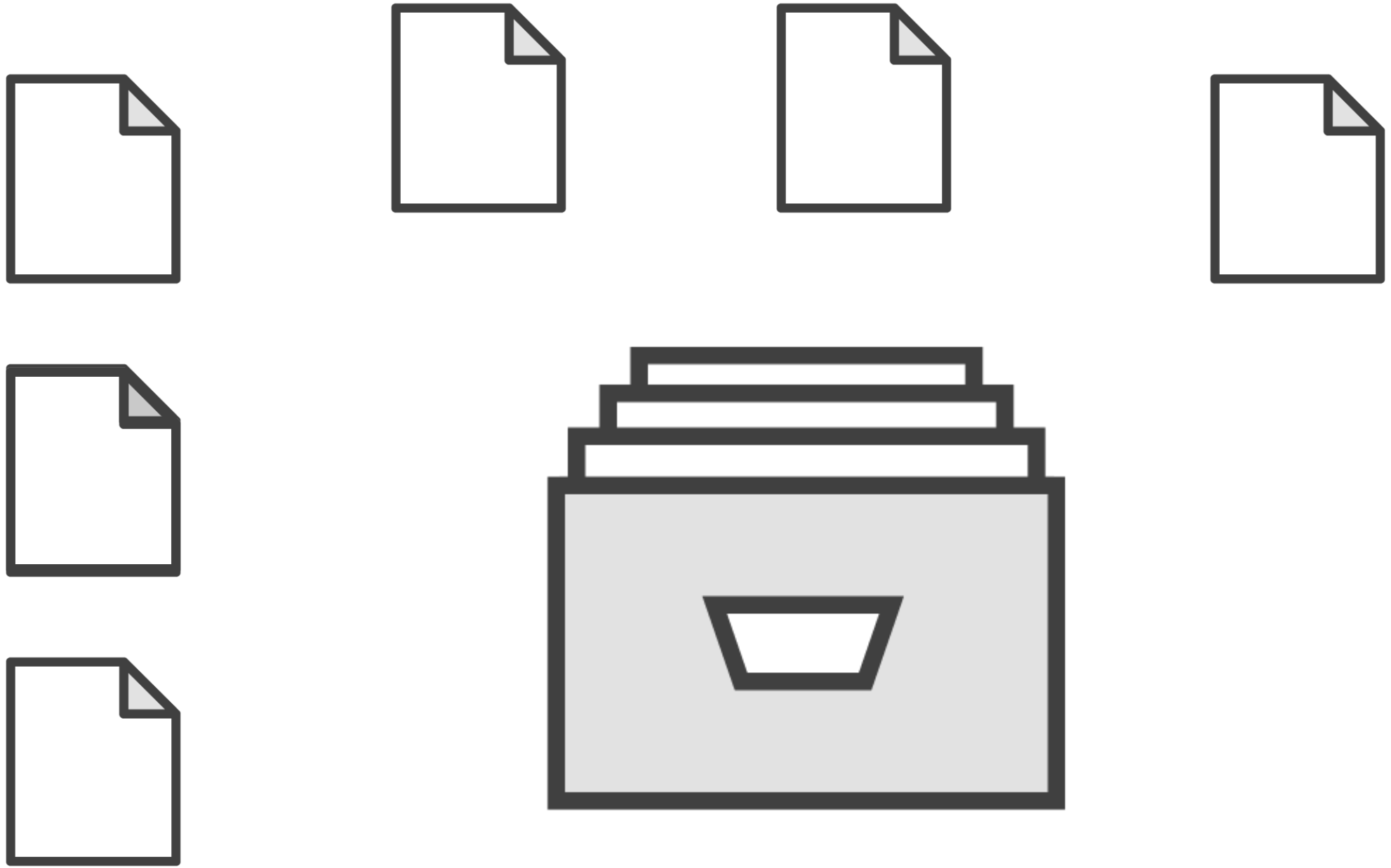


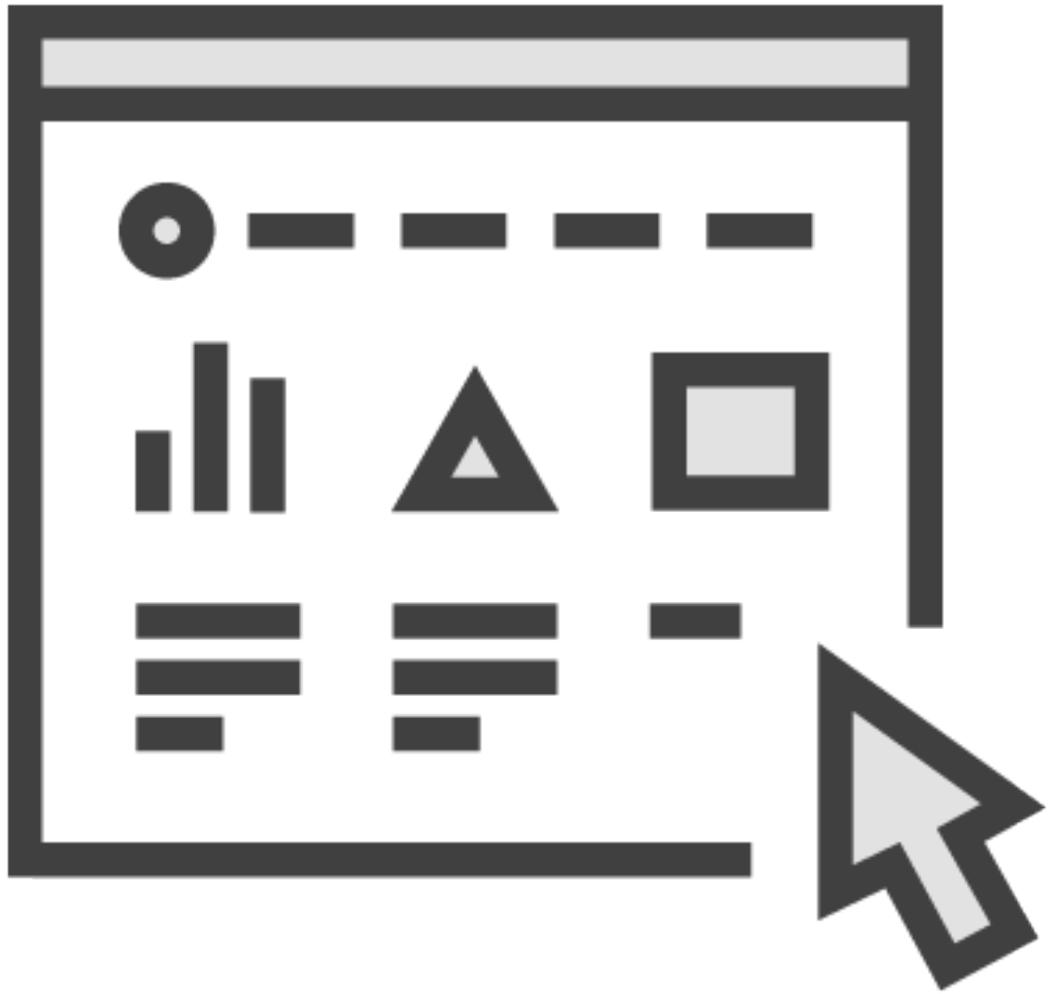




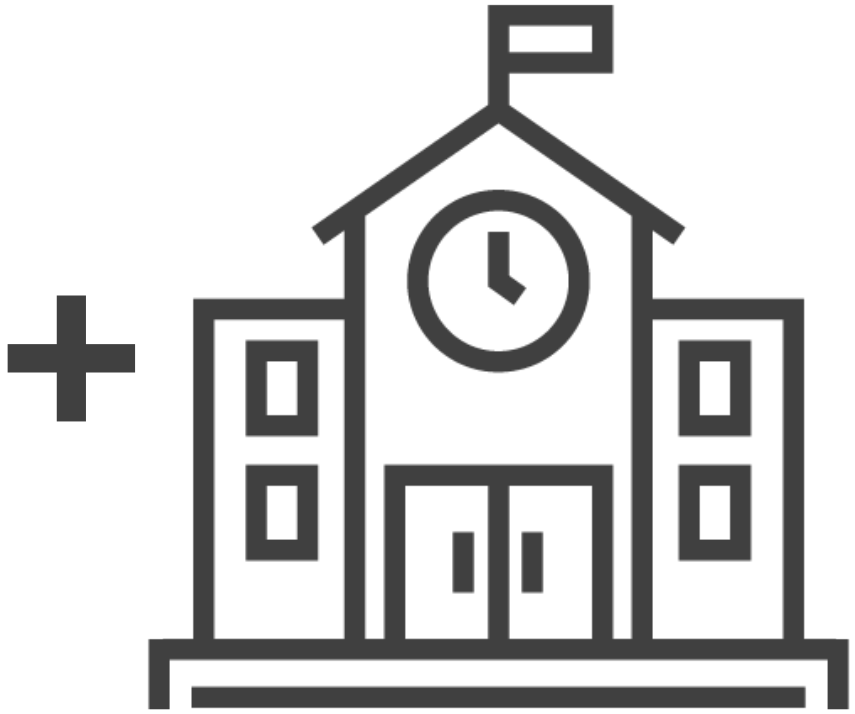


Kafka Uses

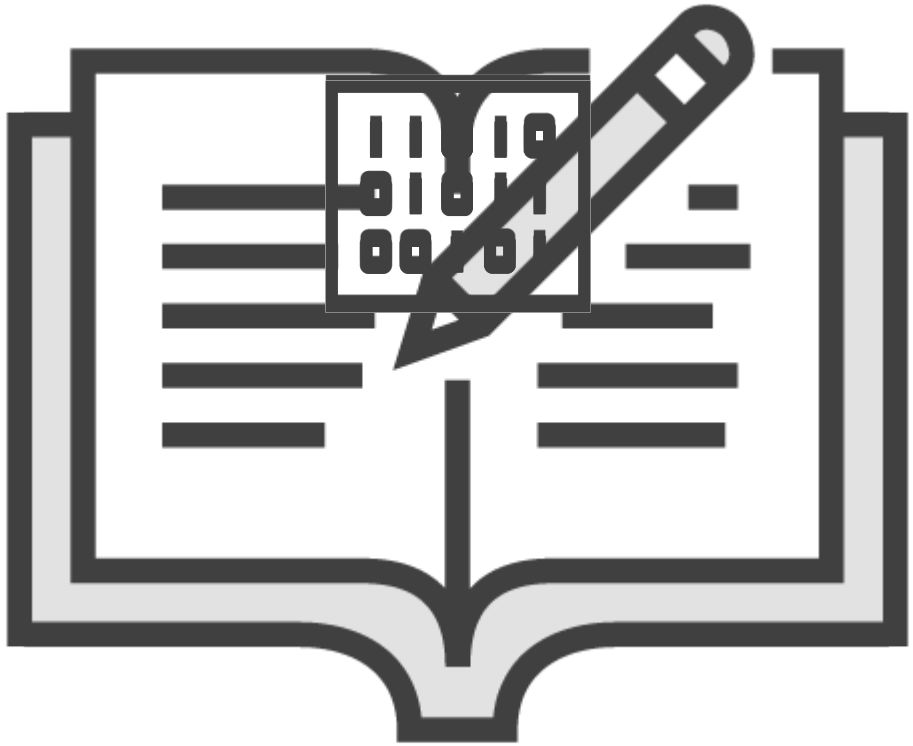


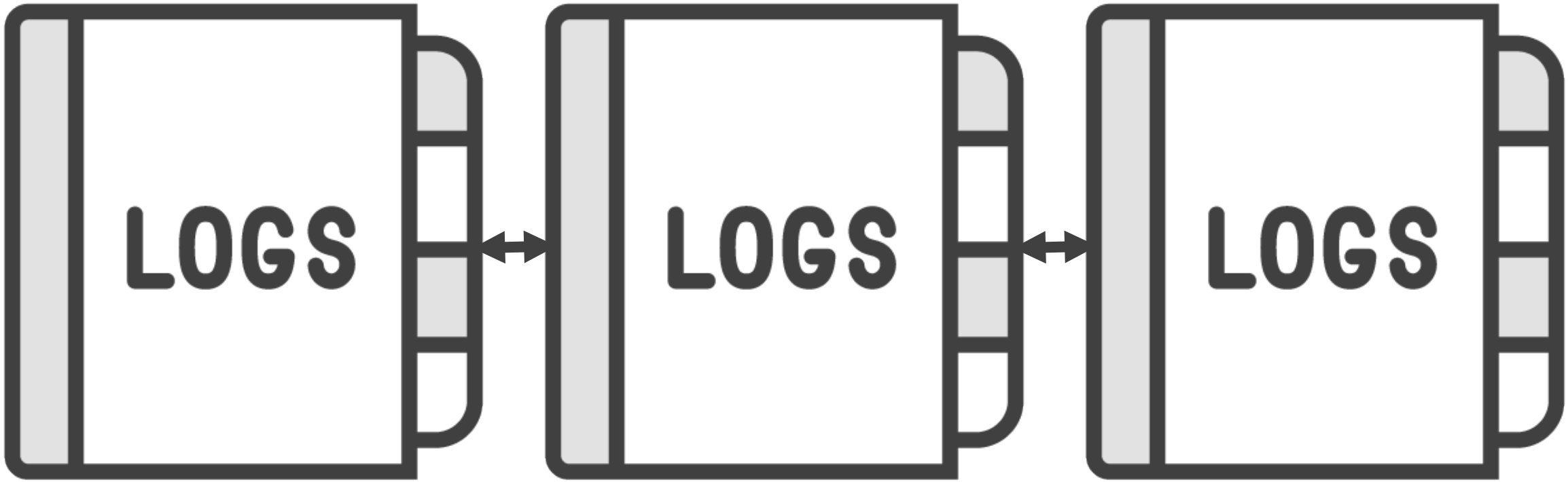


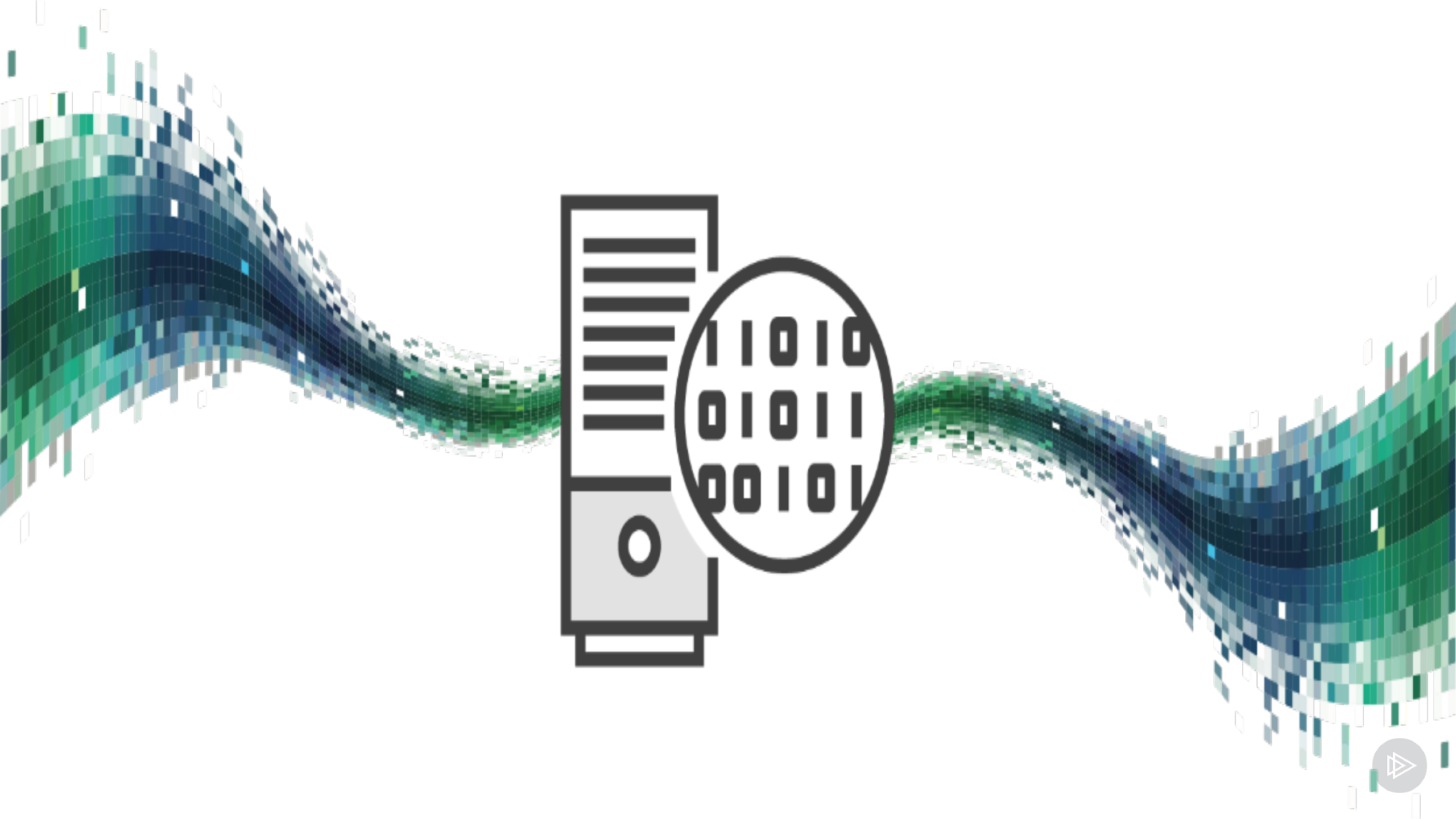
APACHE KAFKA



=



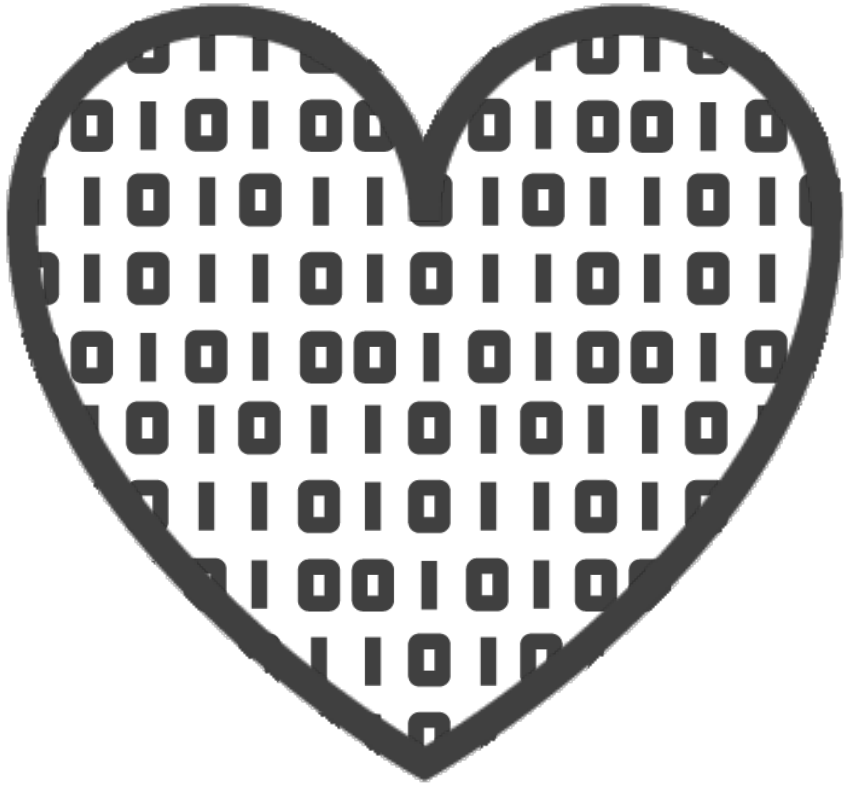

LOGS
LOGS
LOGS

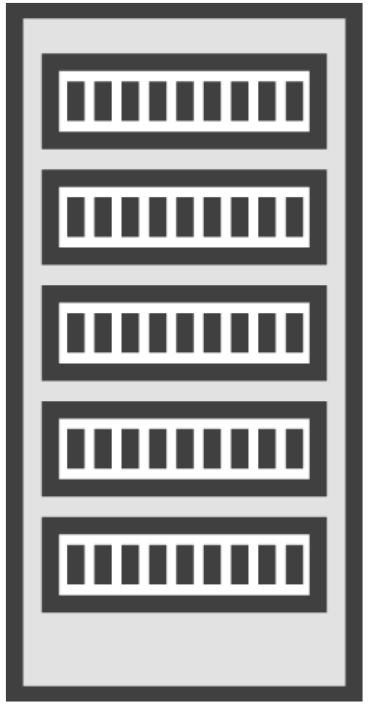


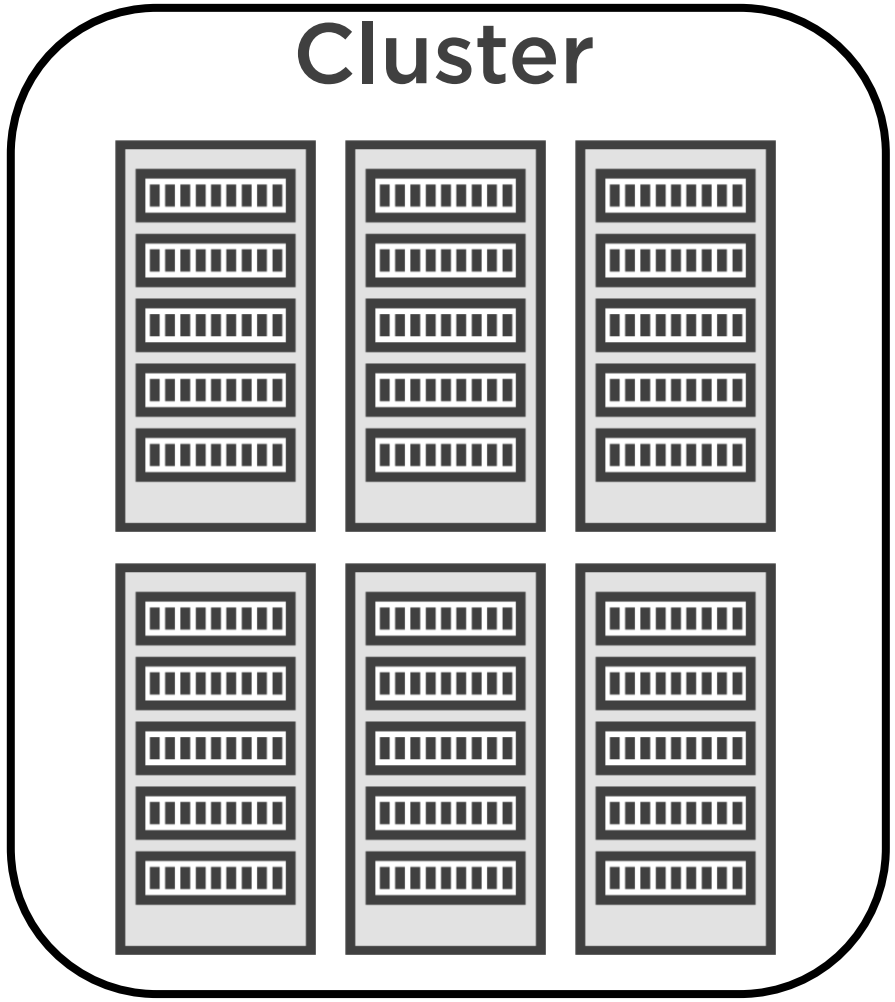

Scala
Java™

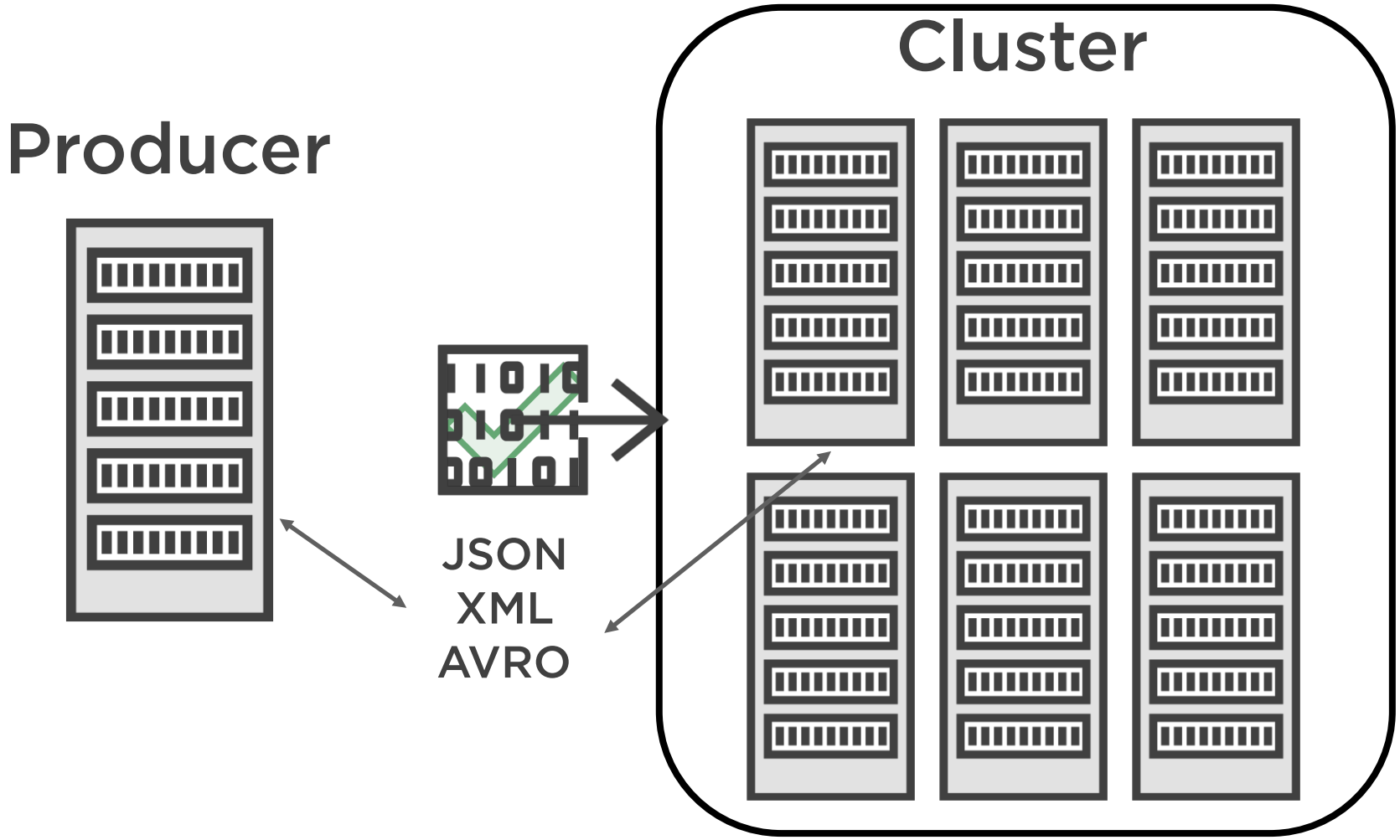
Kafka's Architecture Overview

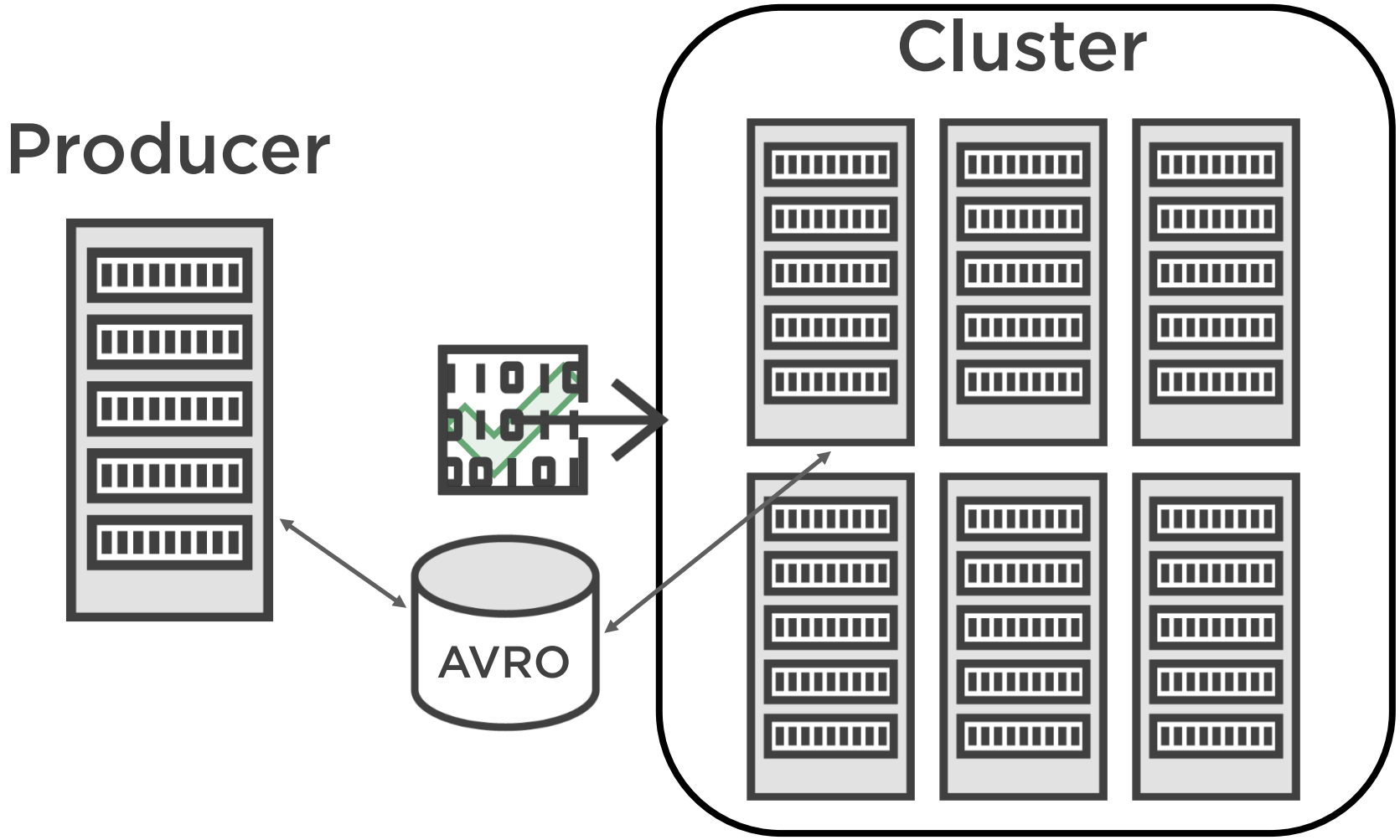

BROKER

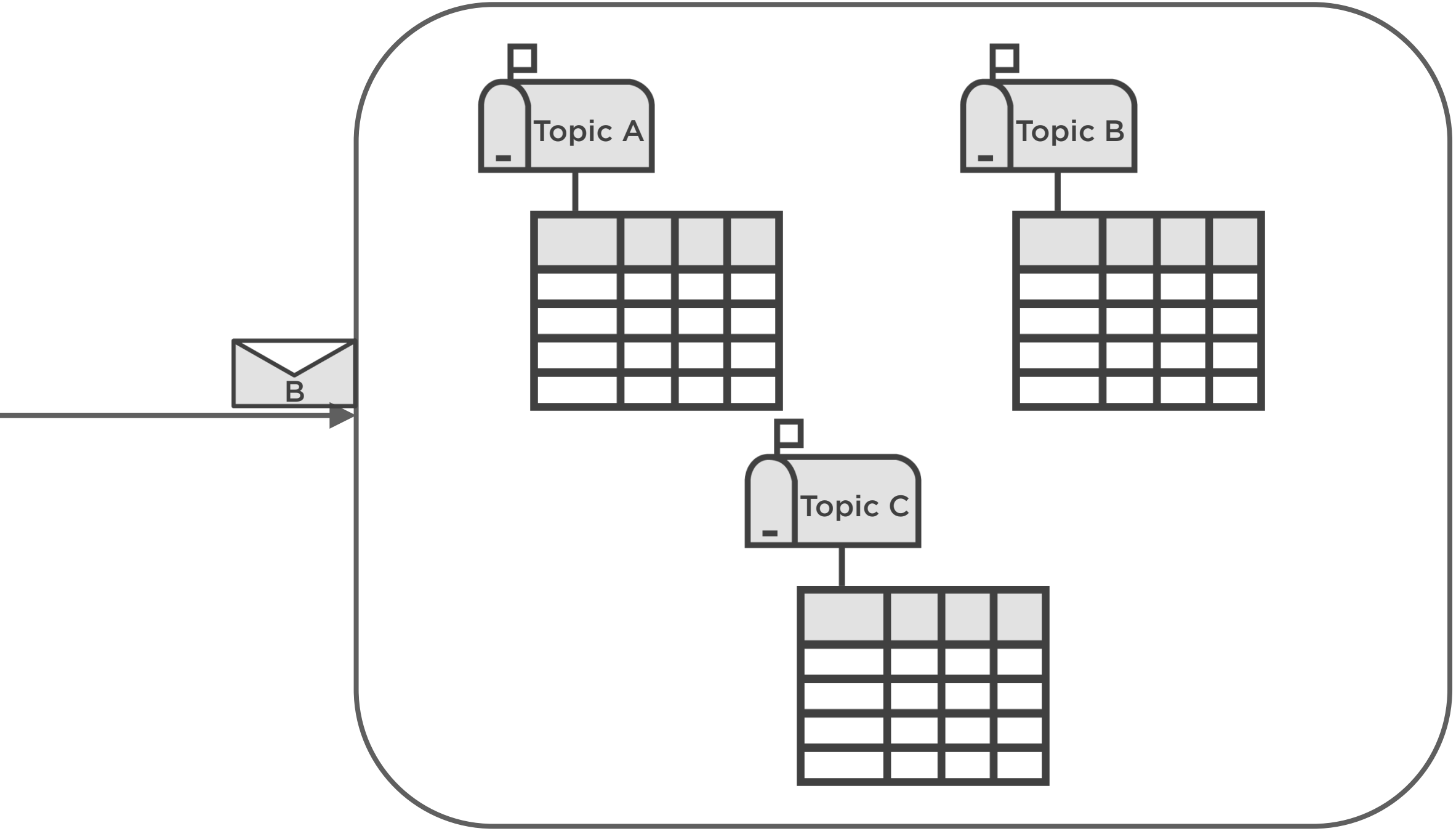


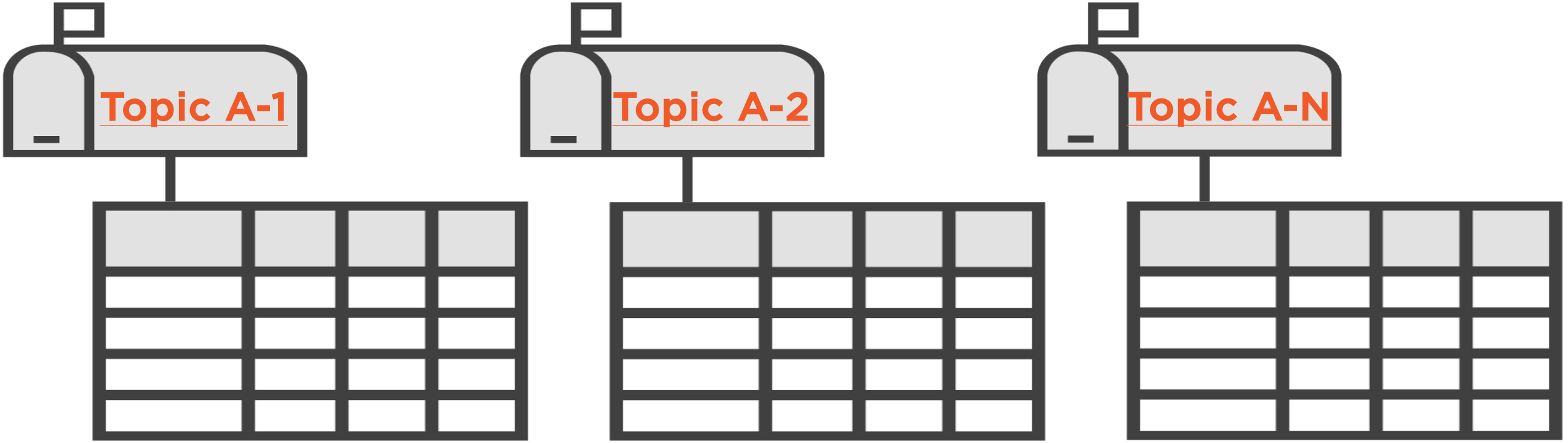


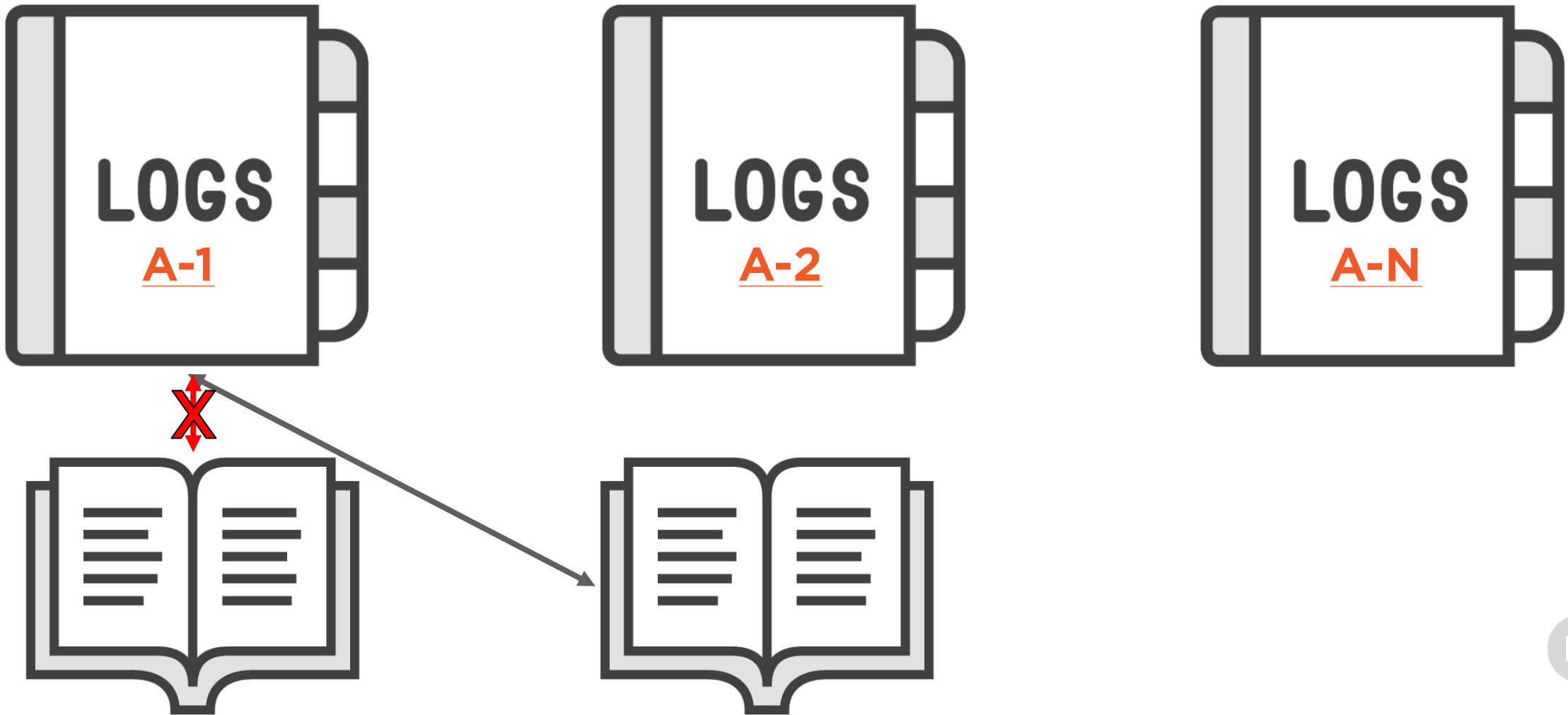

Cluster

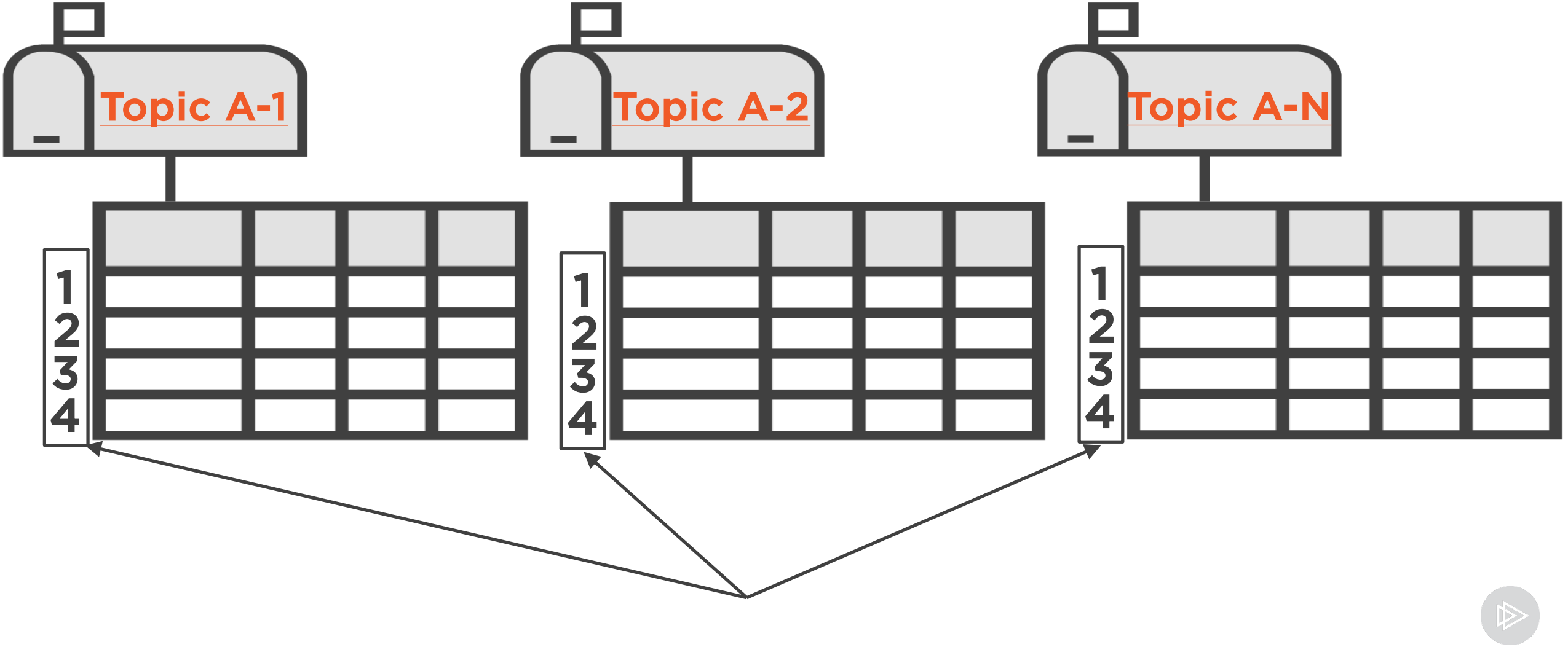

Producer
Cluster
JSON
XML
AVRO

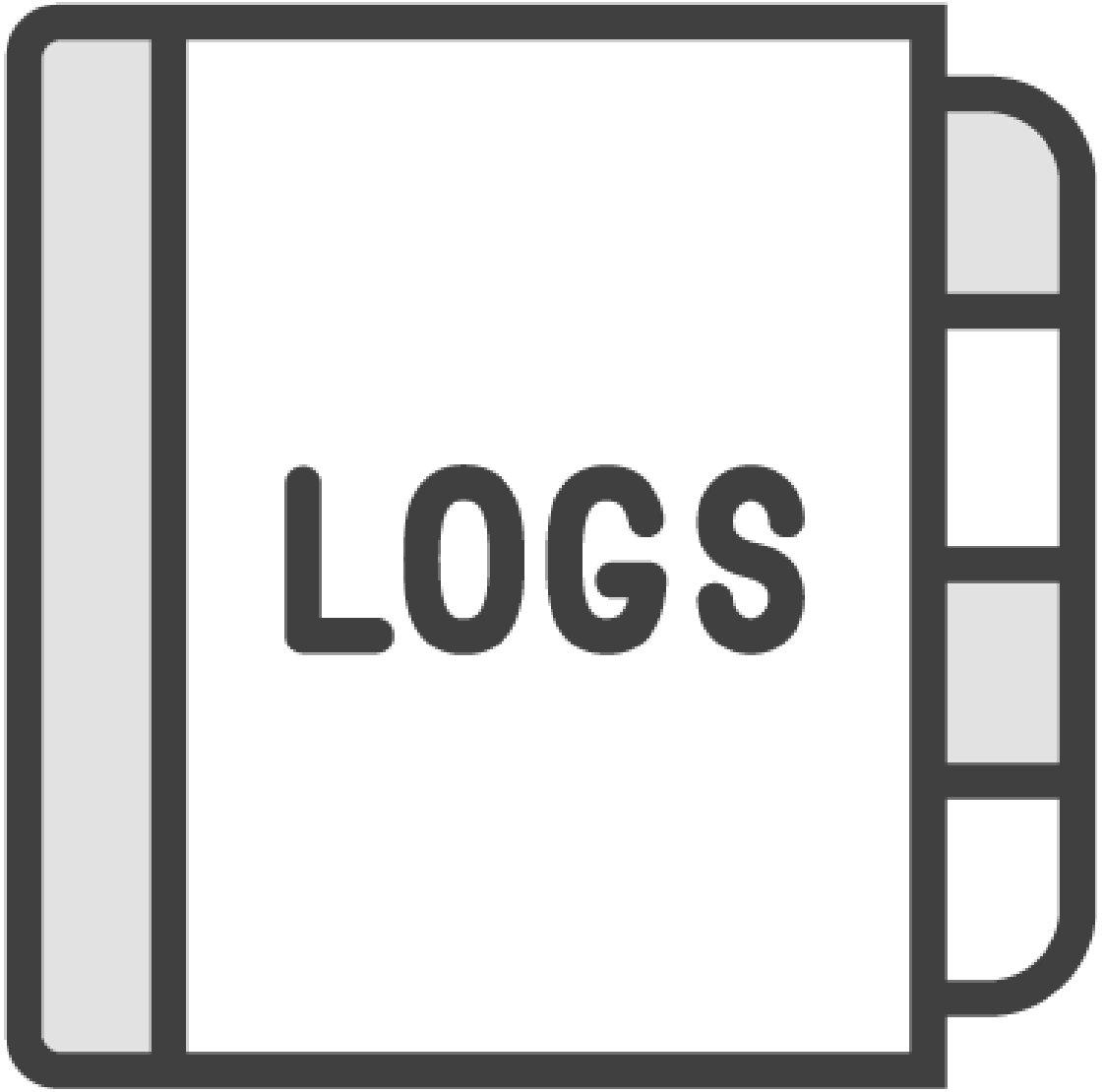

Cluster
Producer
AVRO

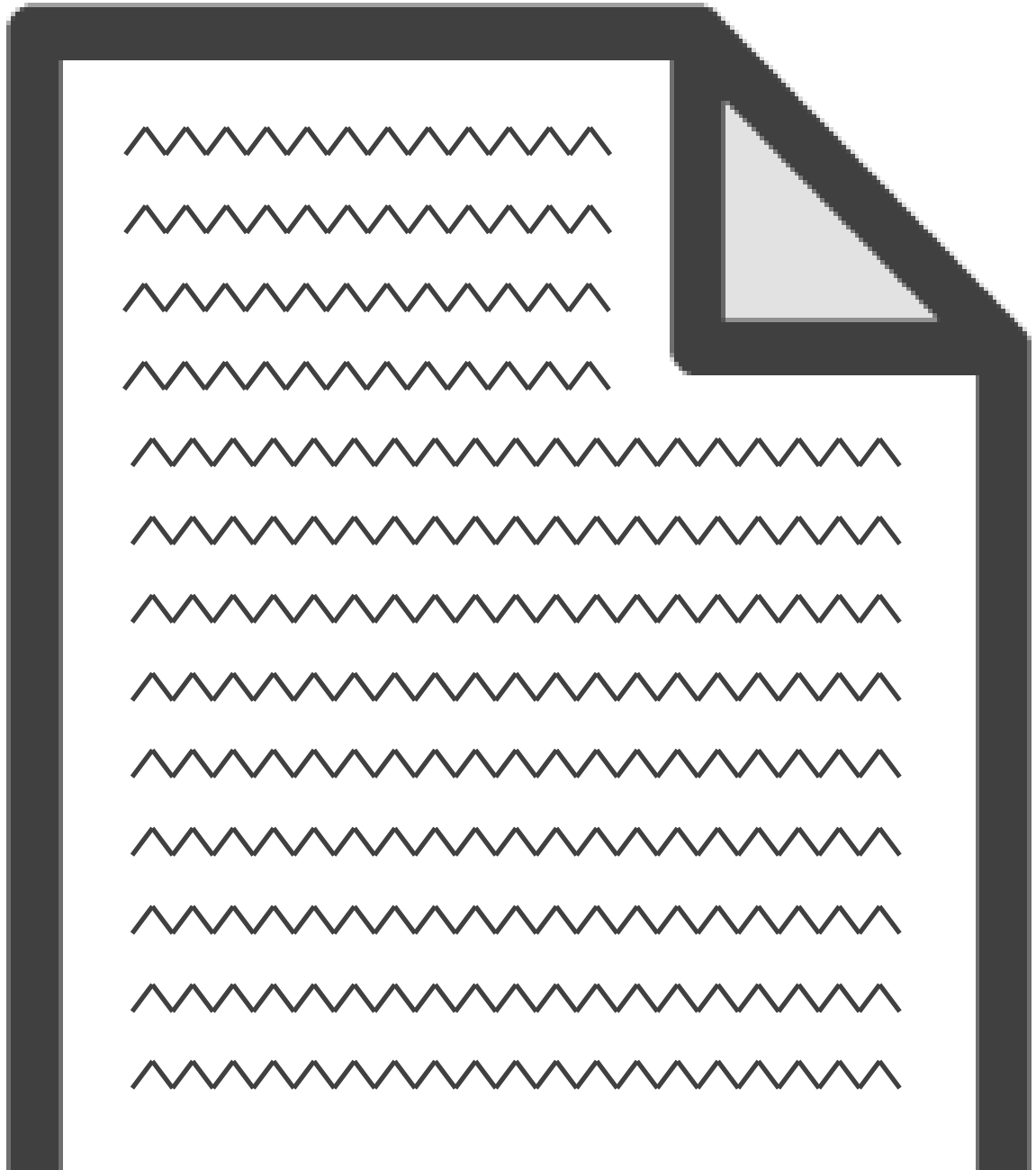

Topic A
Topic B
Topic C
B

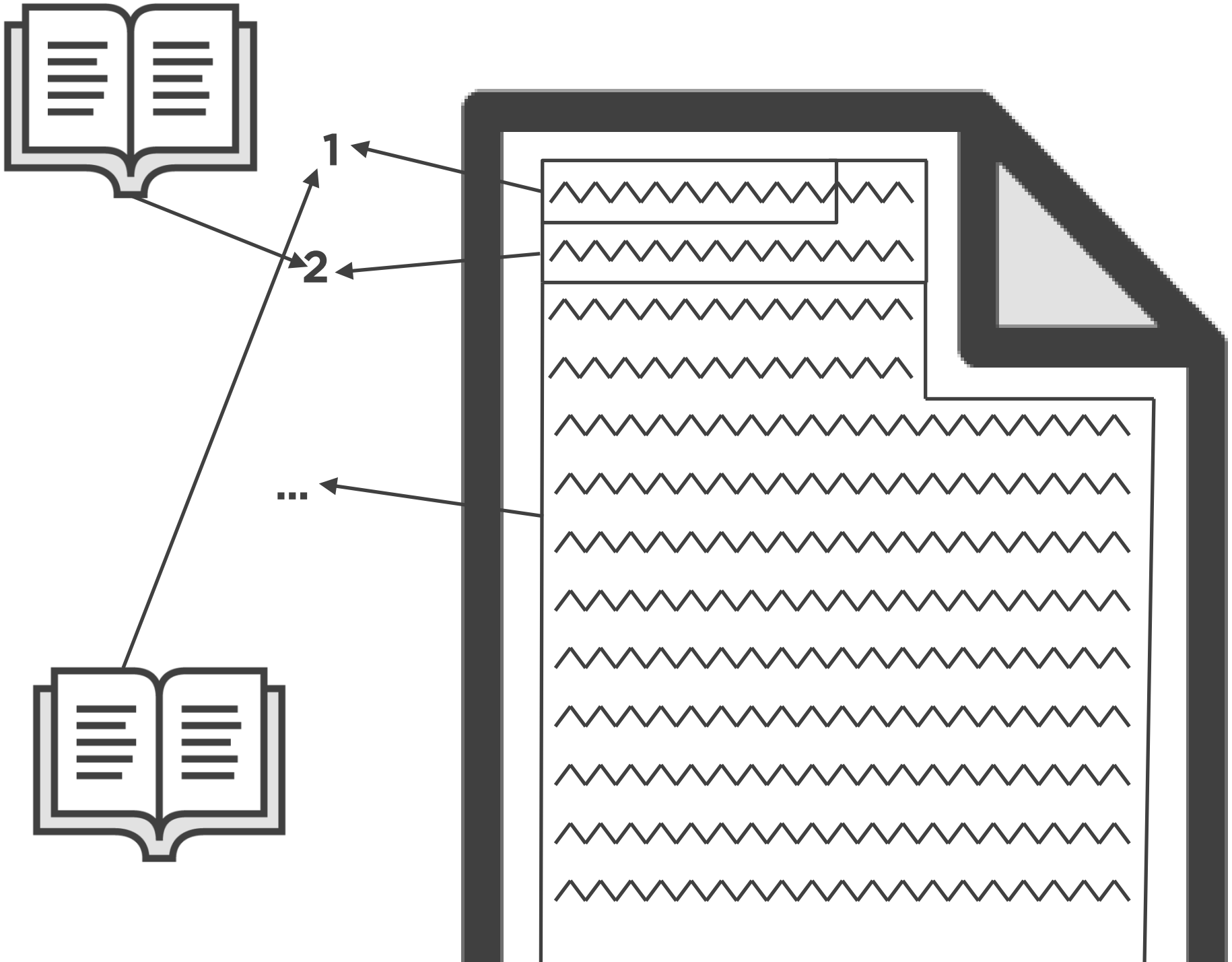

Topic A-1
Topic A-2
Topic A-N

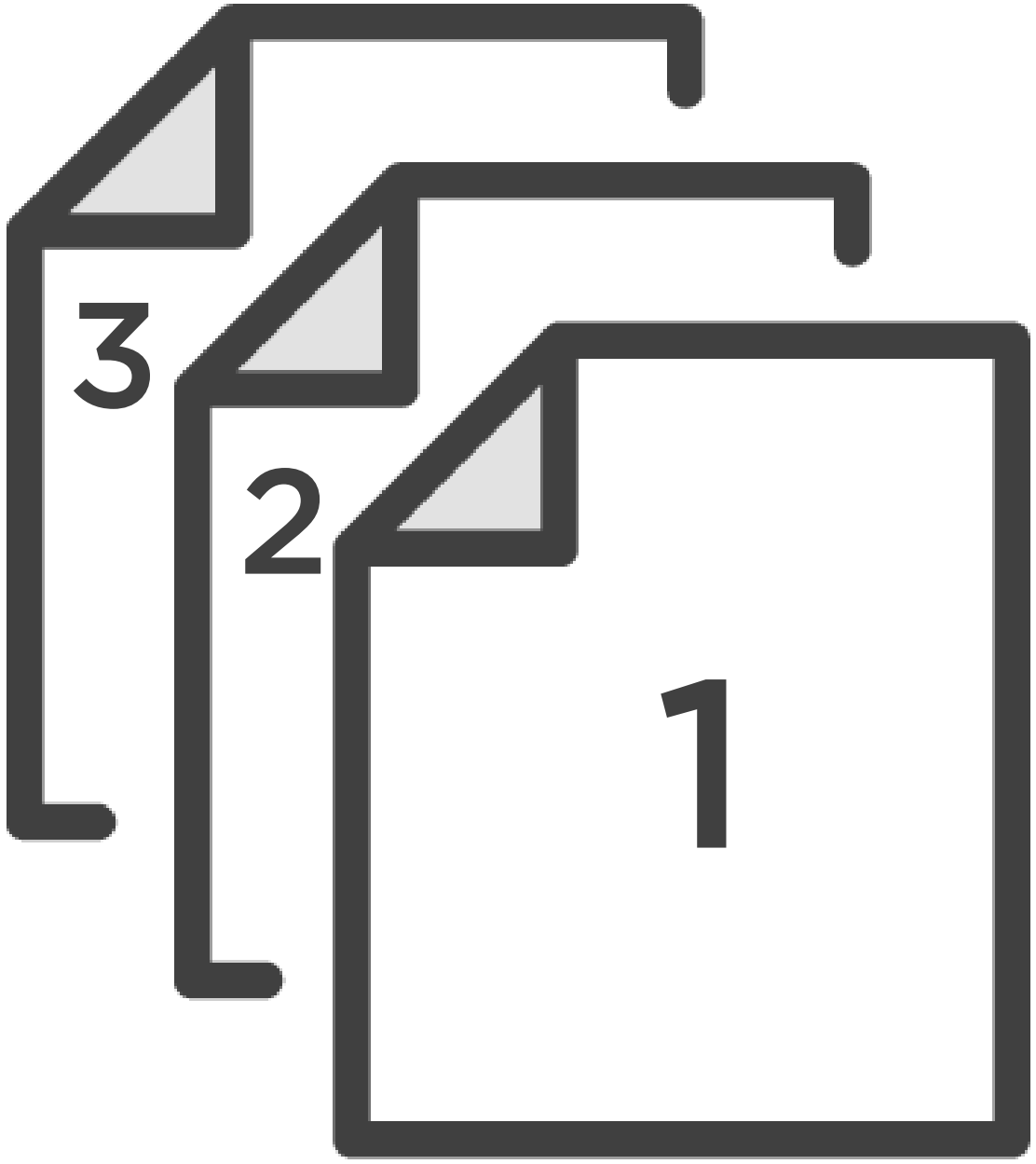

LOGS
A-1
LOGS
A-2
LOGS
A-N


Topic A-1
Topic A-2
Topic A-N
1
2
3
4
1
2
3
4
1
2
3
4


LOGS



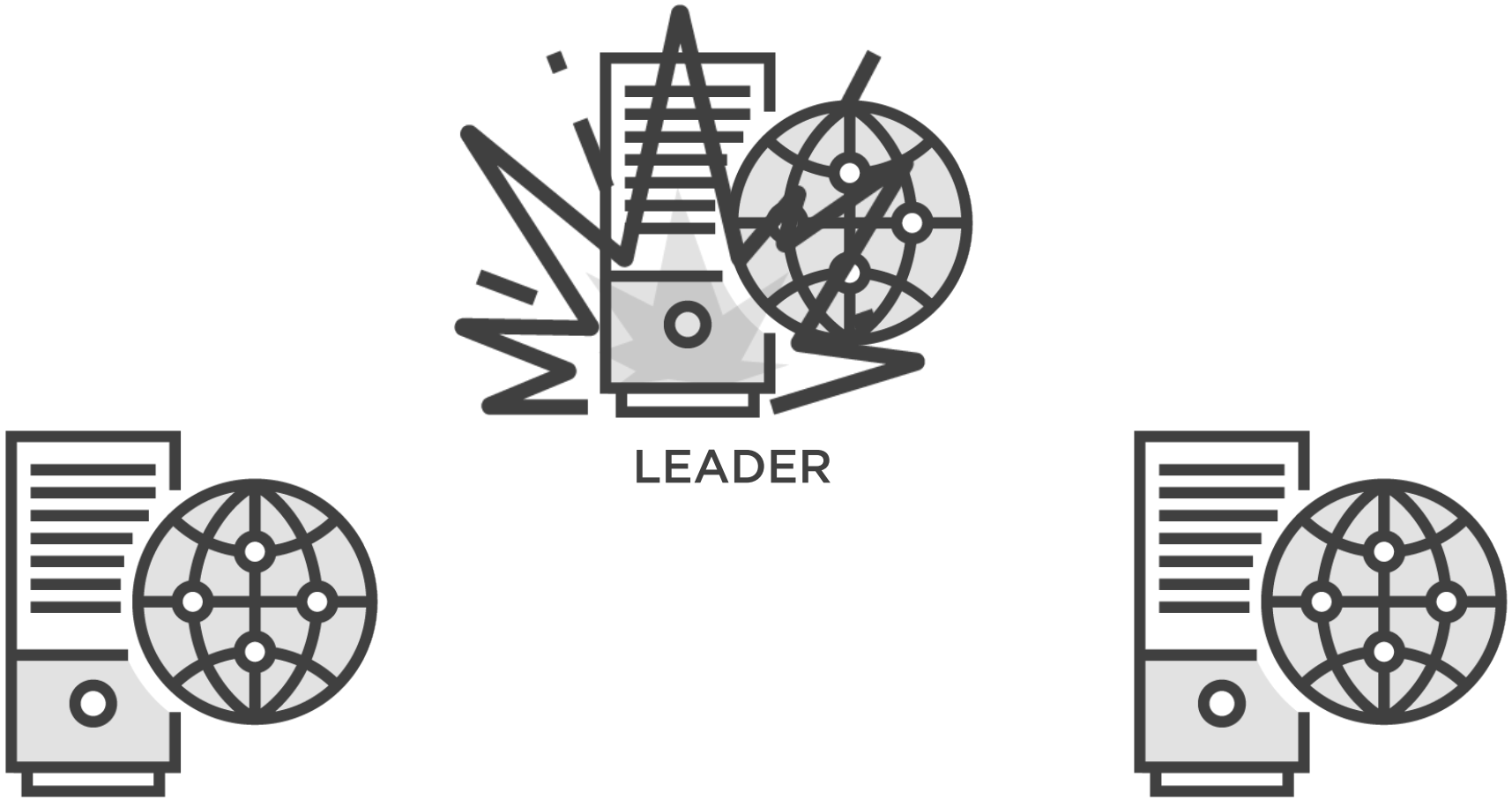

1
2
...


3
2
1

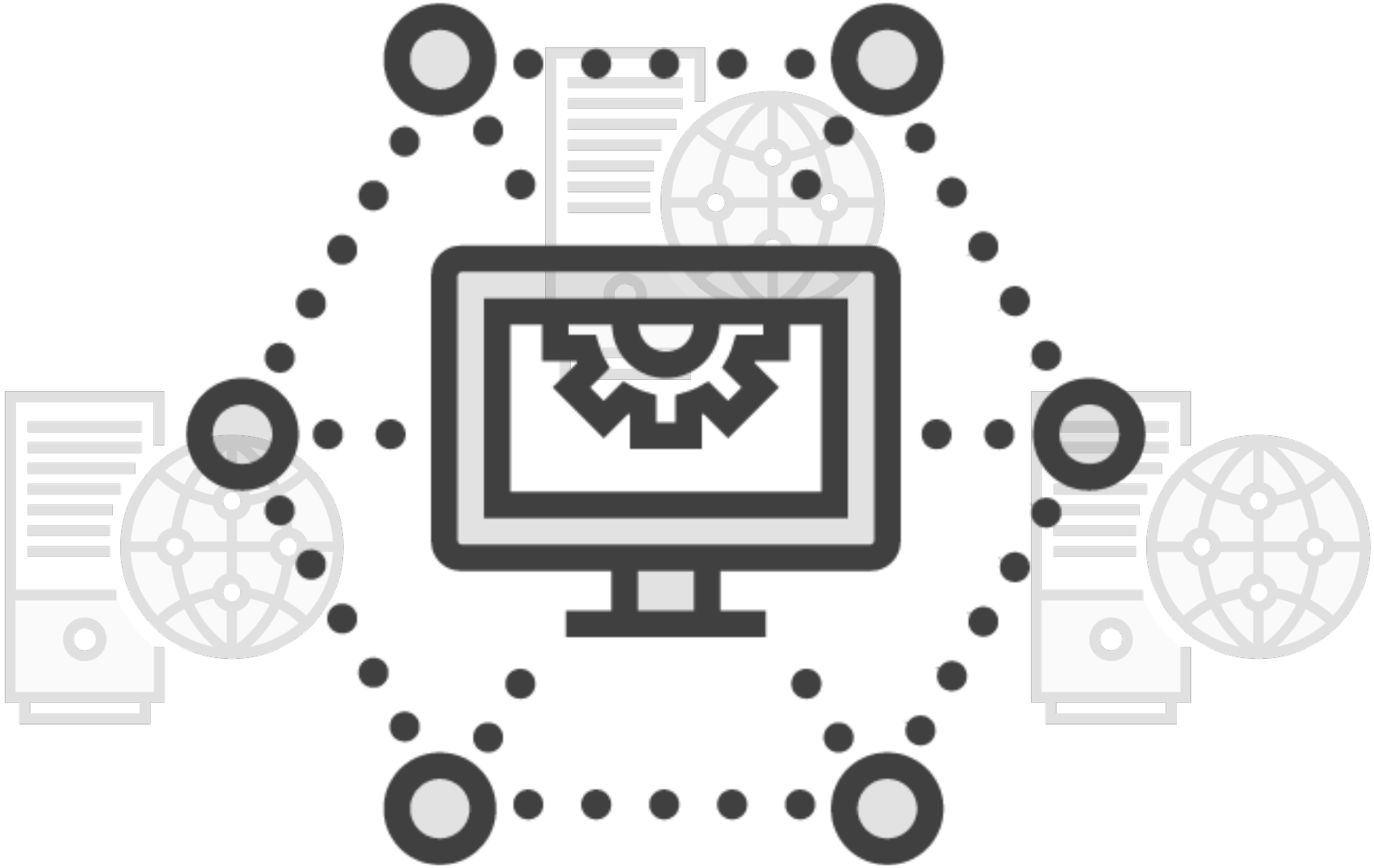
Apache ZooKeeper

...a centralized service for maintaining configuration information, naming, providing distributed synchronization, and providing group services.

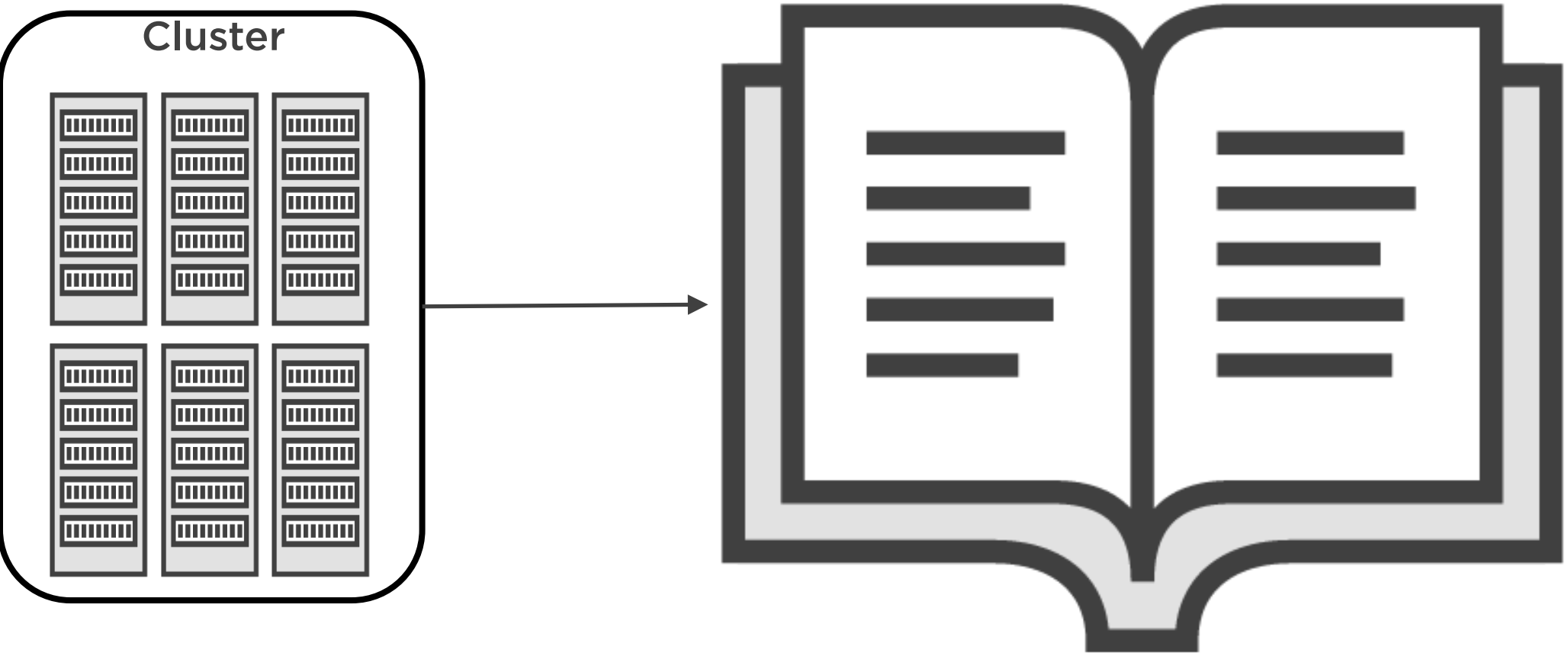
Apache ZooKeeper



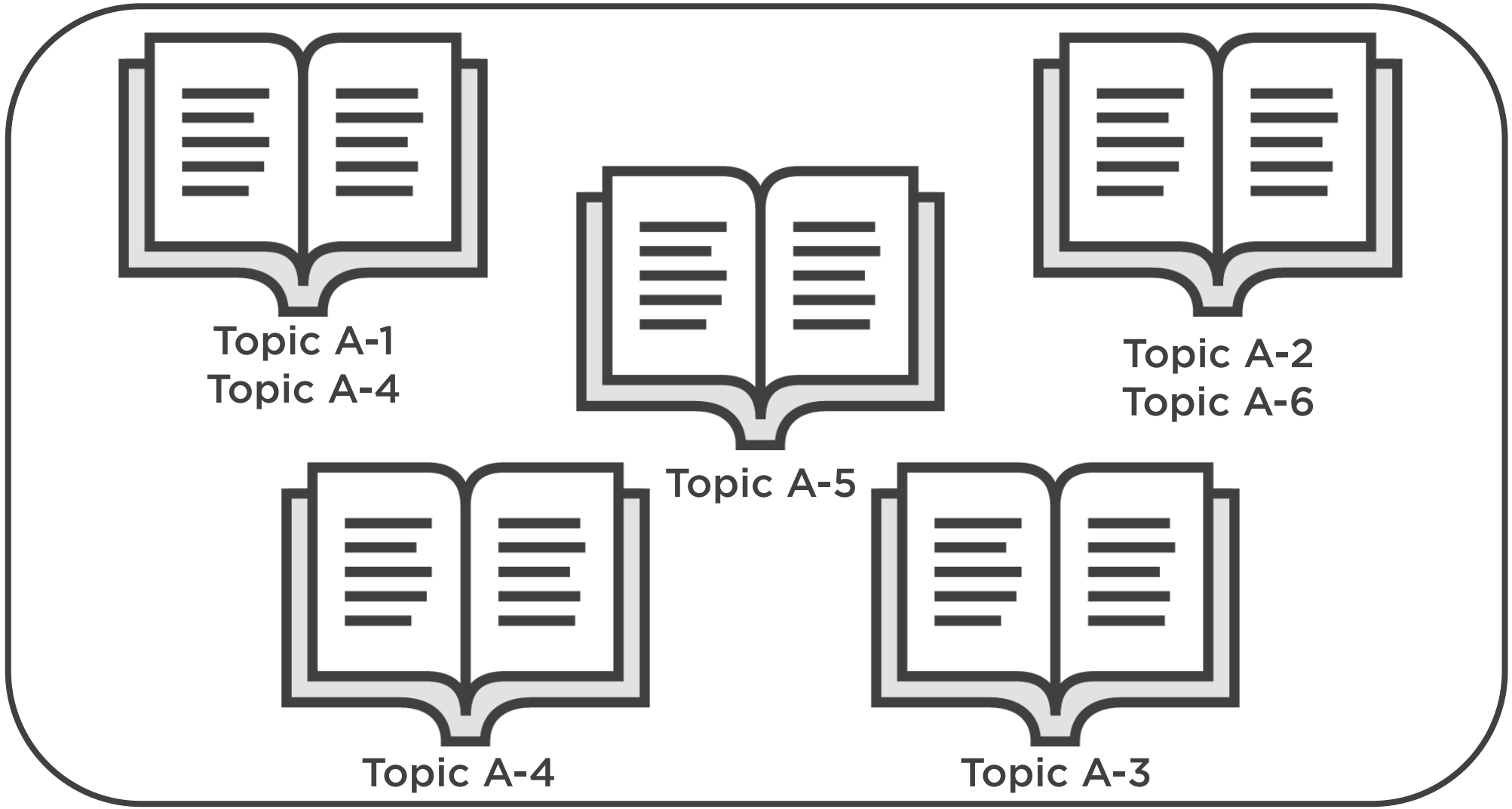
Apache ZooKeeper



Consumer

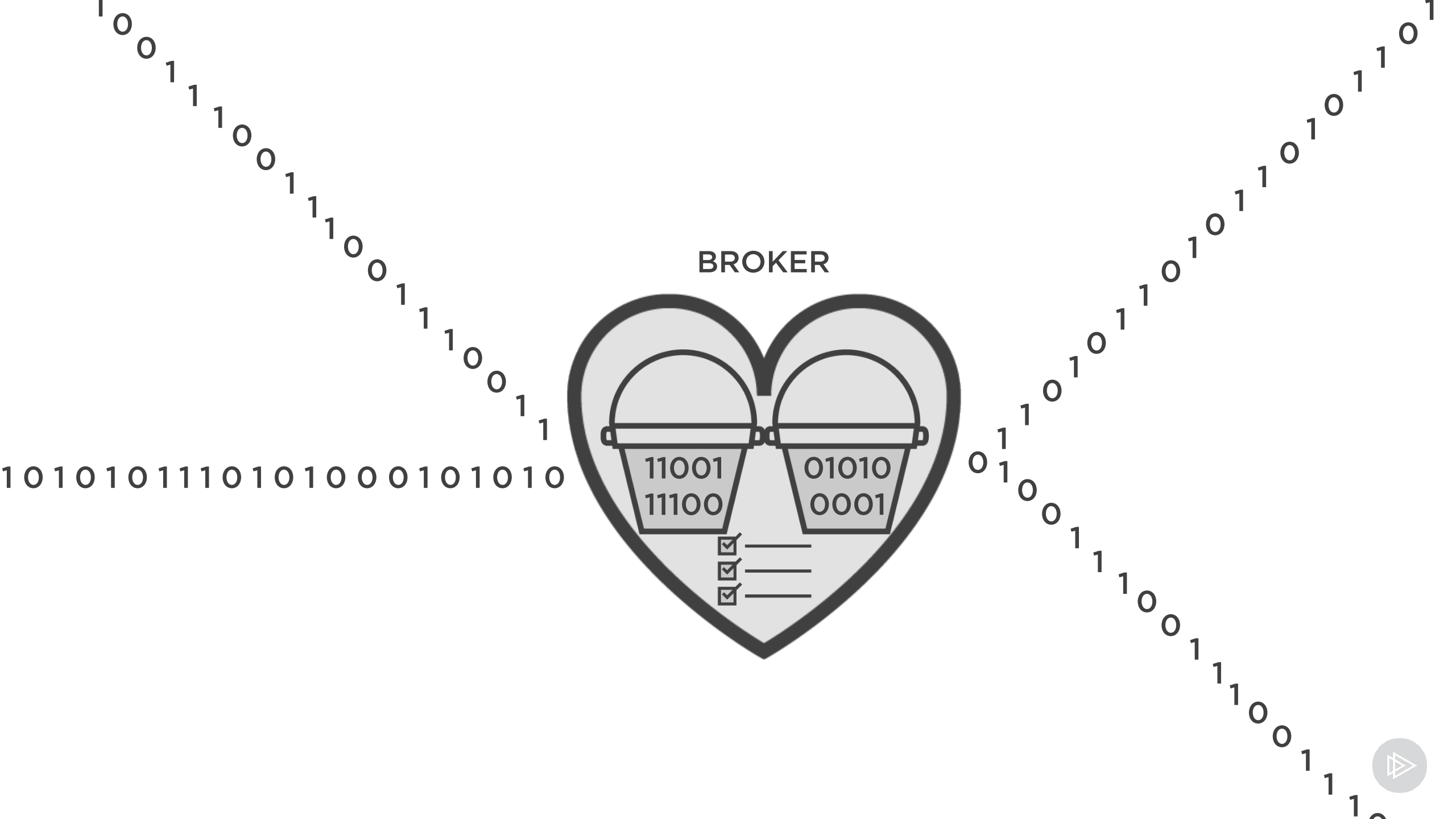


Consumer Group

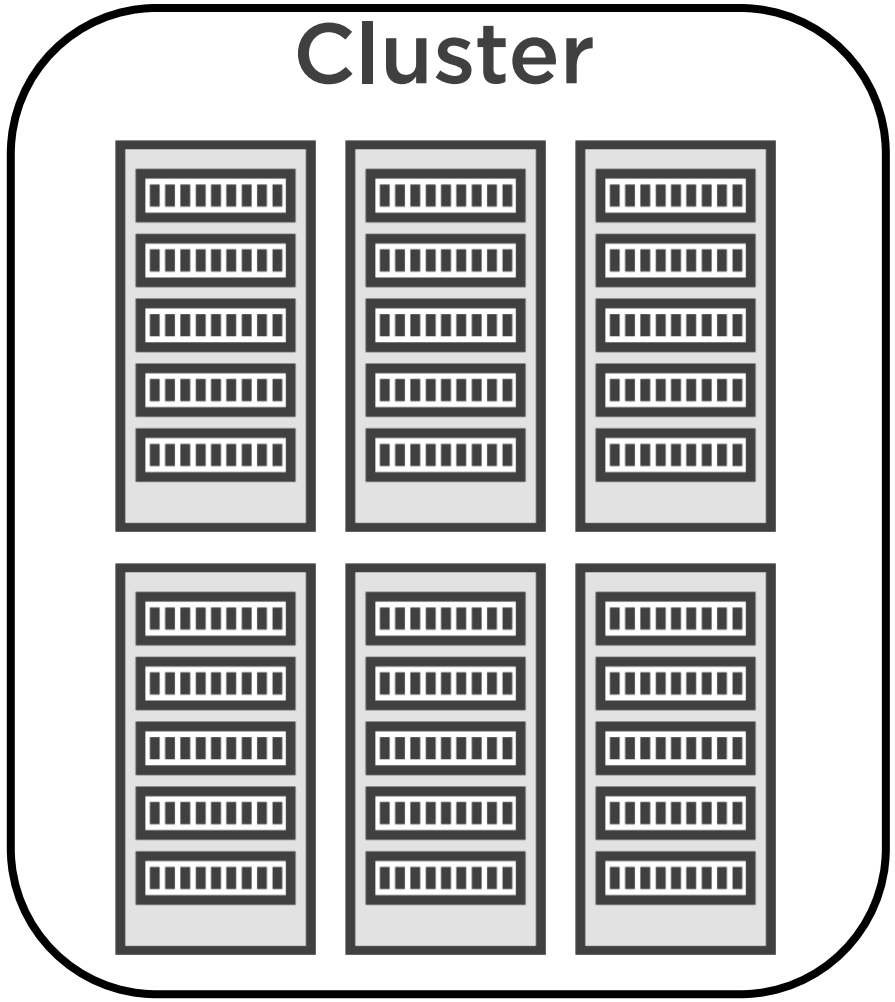


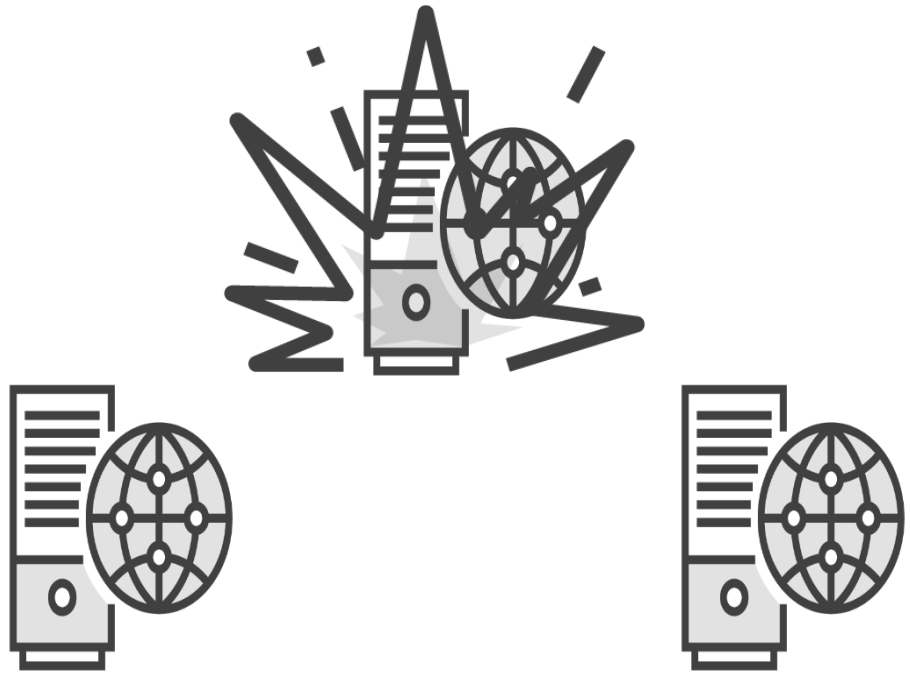

APACHE
kafka®

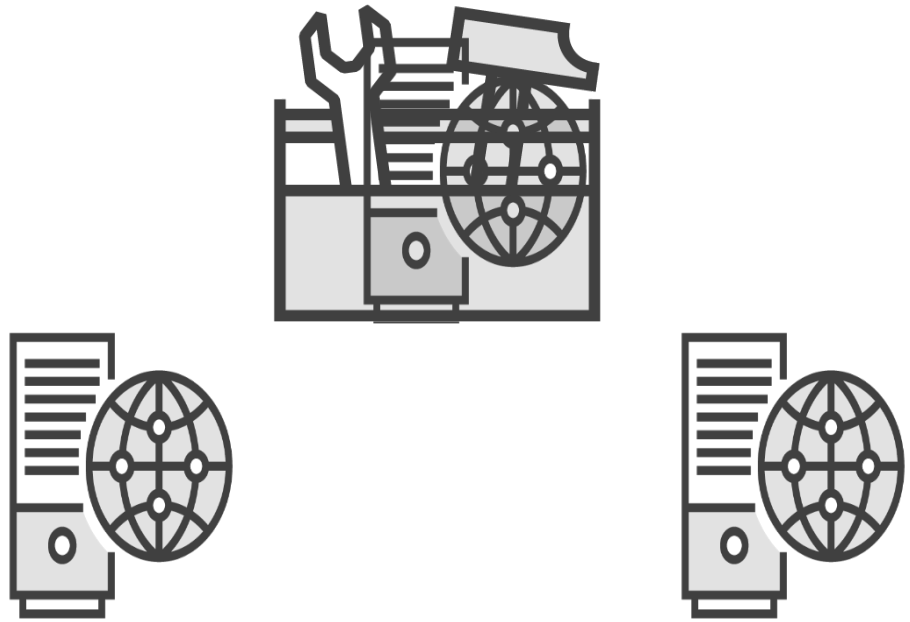
Kafka's Heart: The Broker


BROKER
11001
11100
01010
0001


BROKER

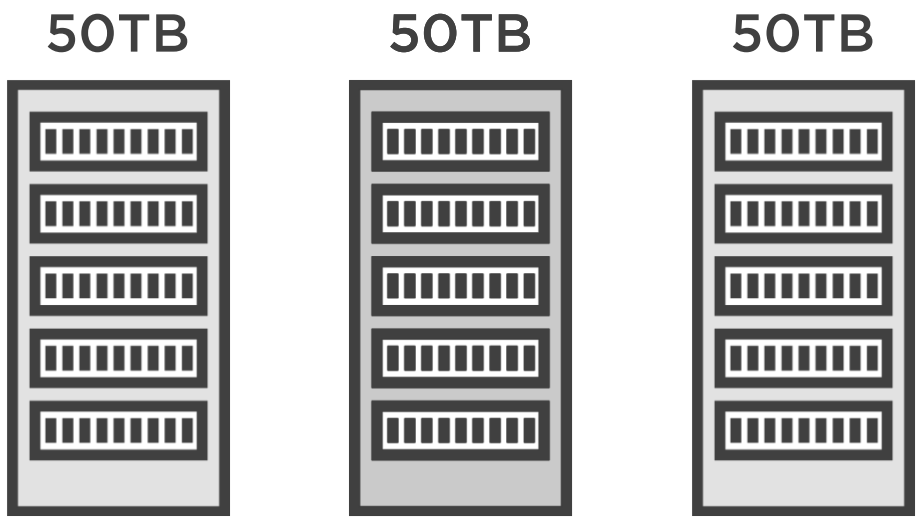

Cluster



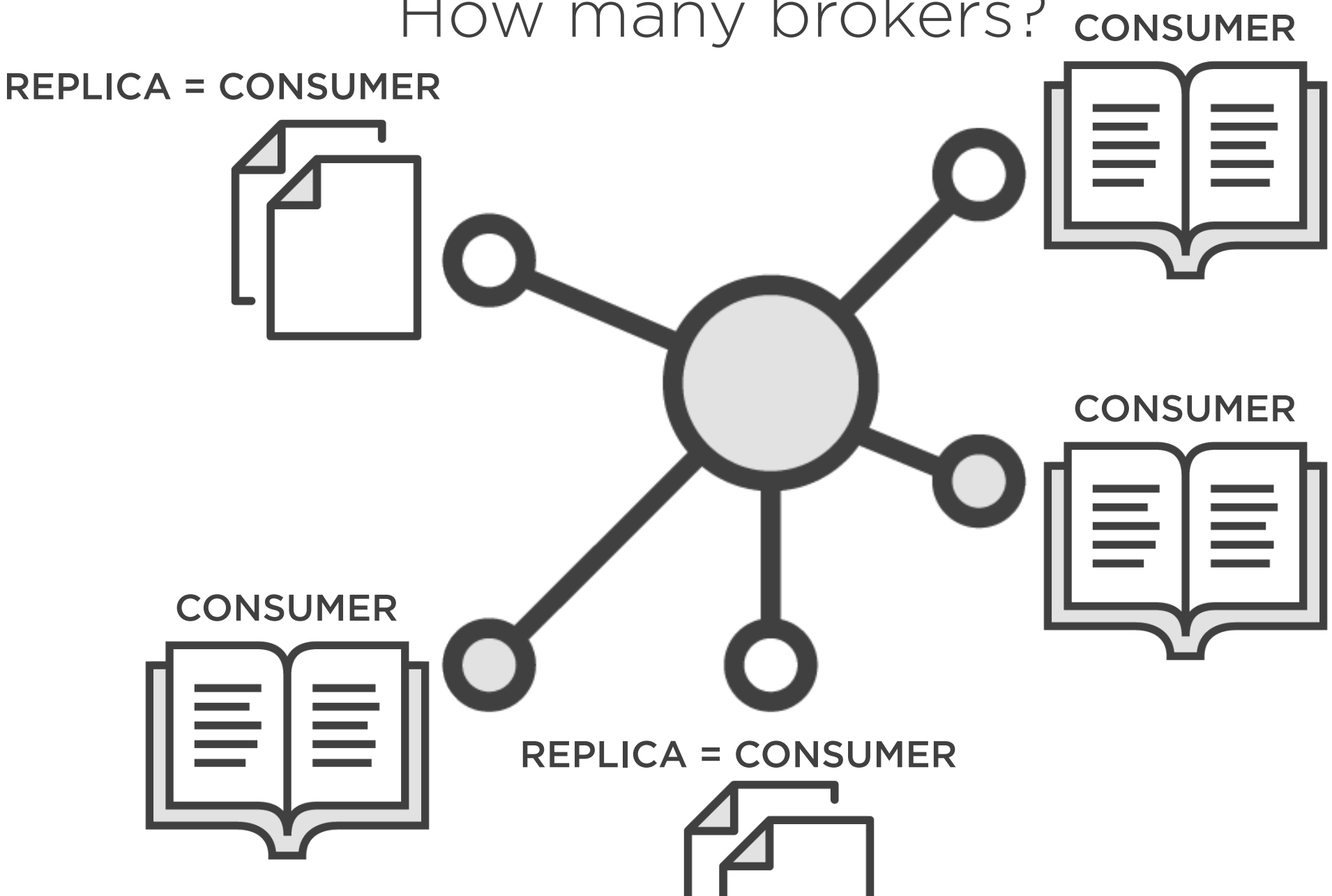


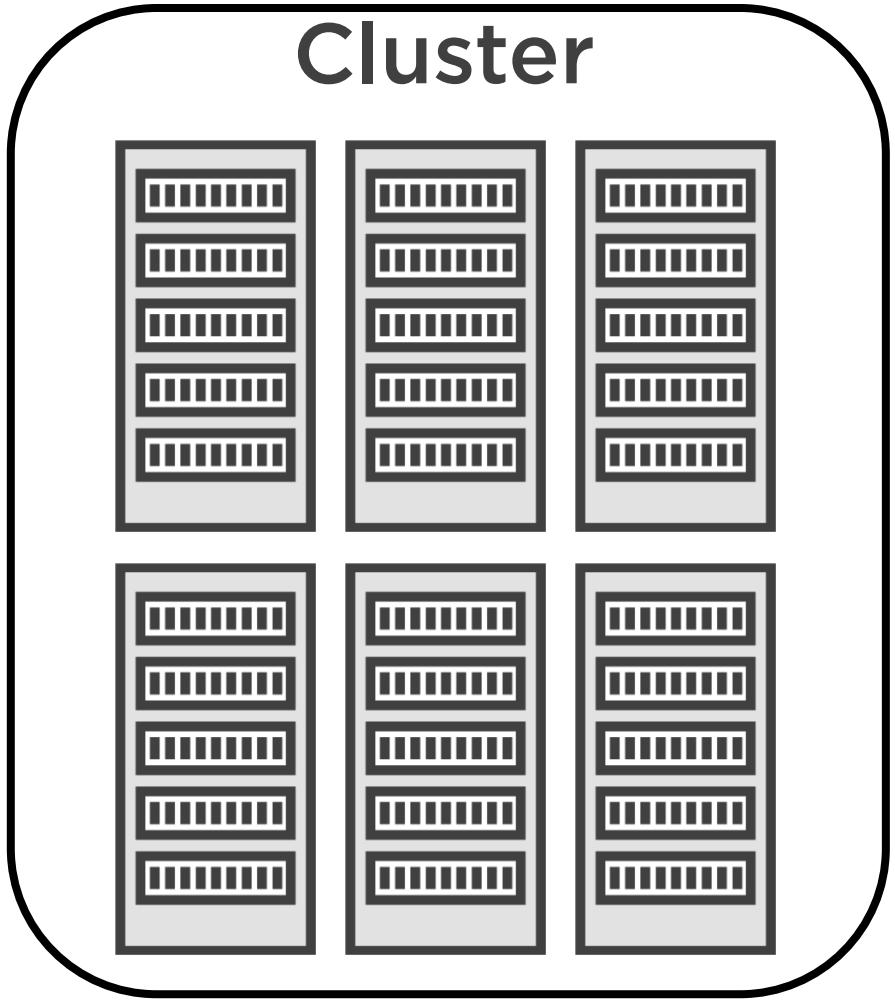
How many brokers?

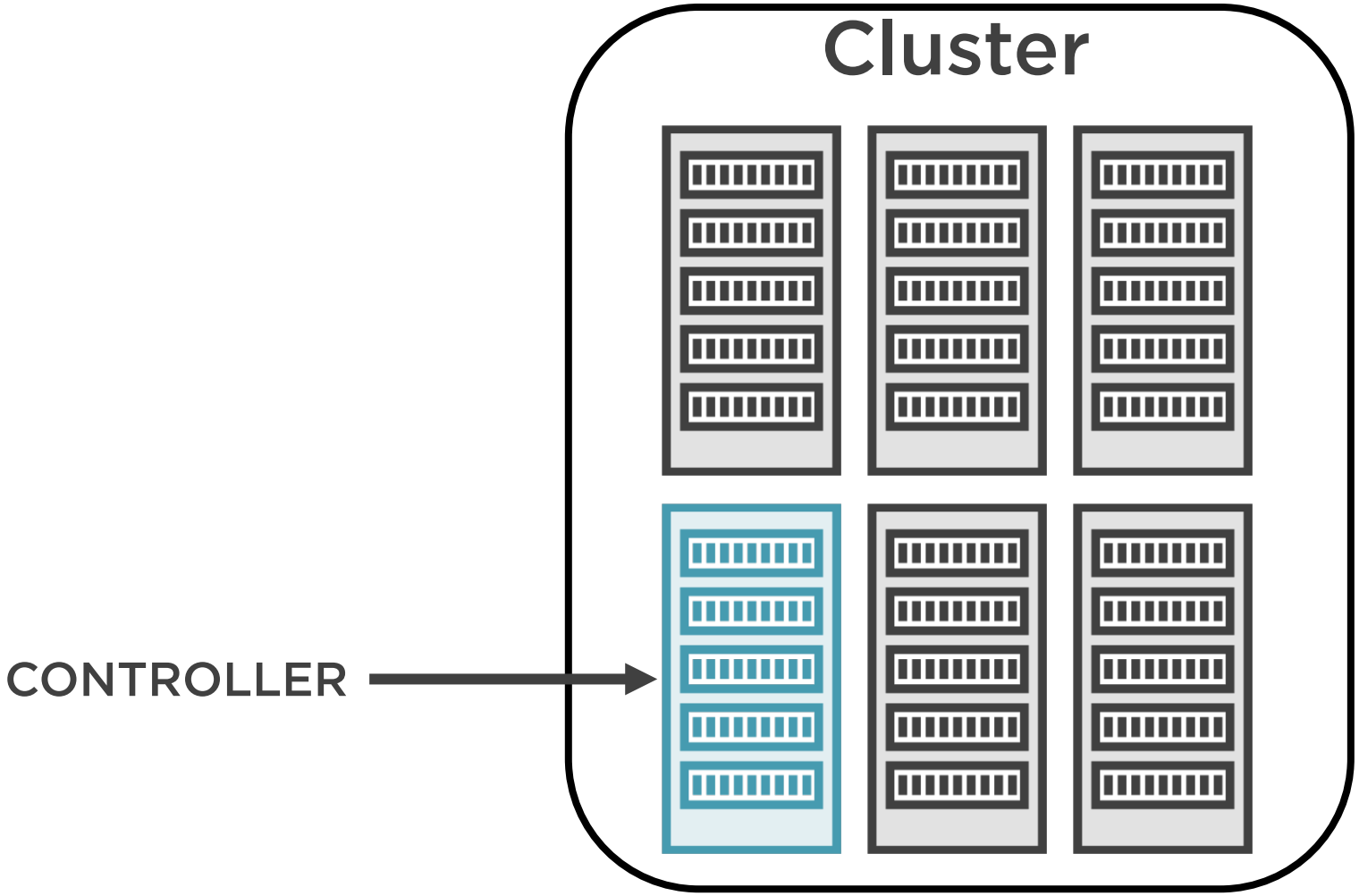
How many brokers?



How many brokers?

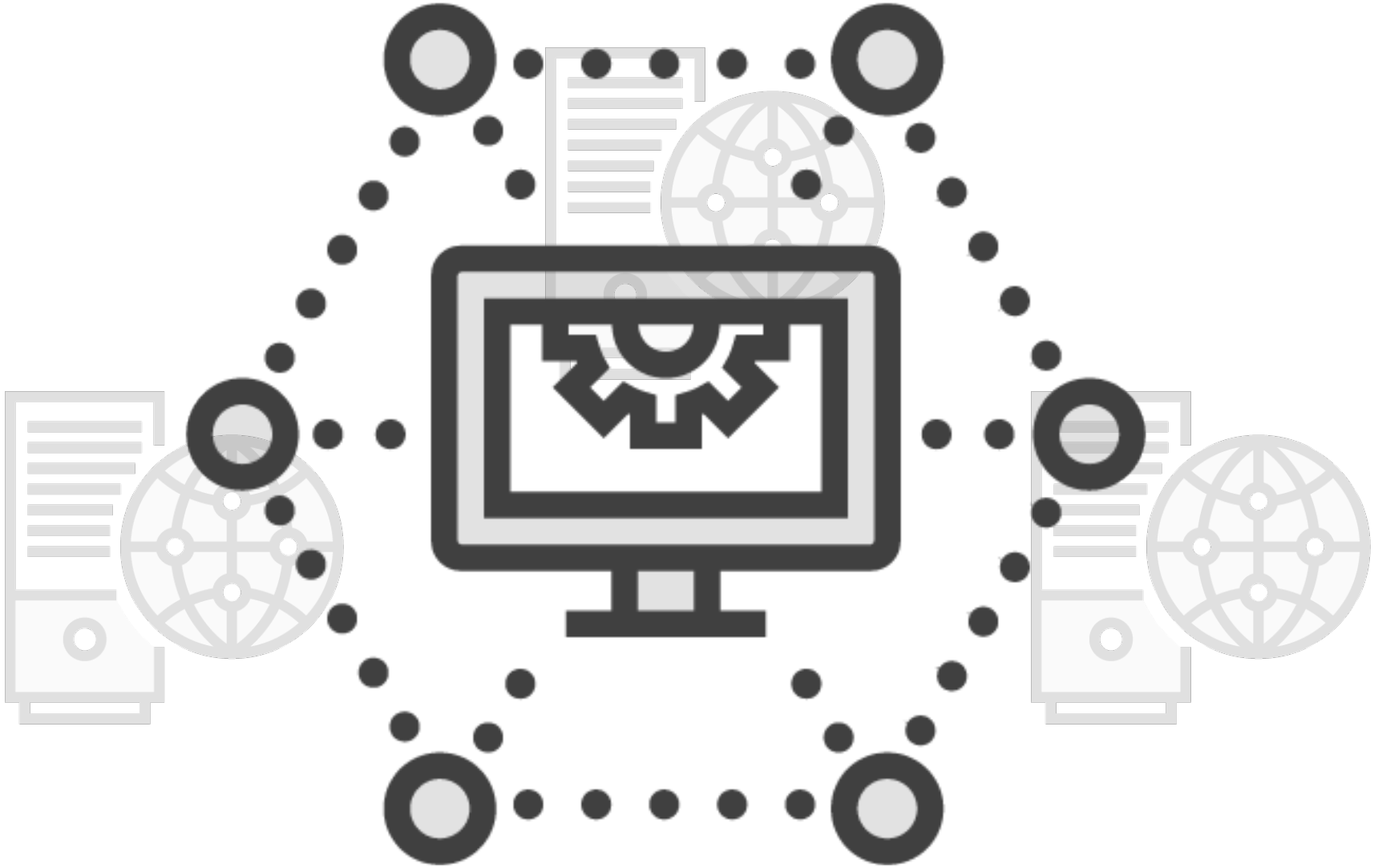



Cluster


Cluster
CONTROLLER

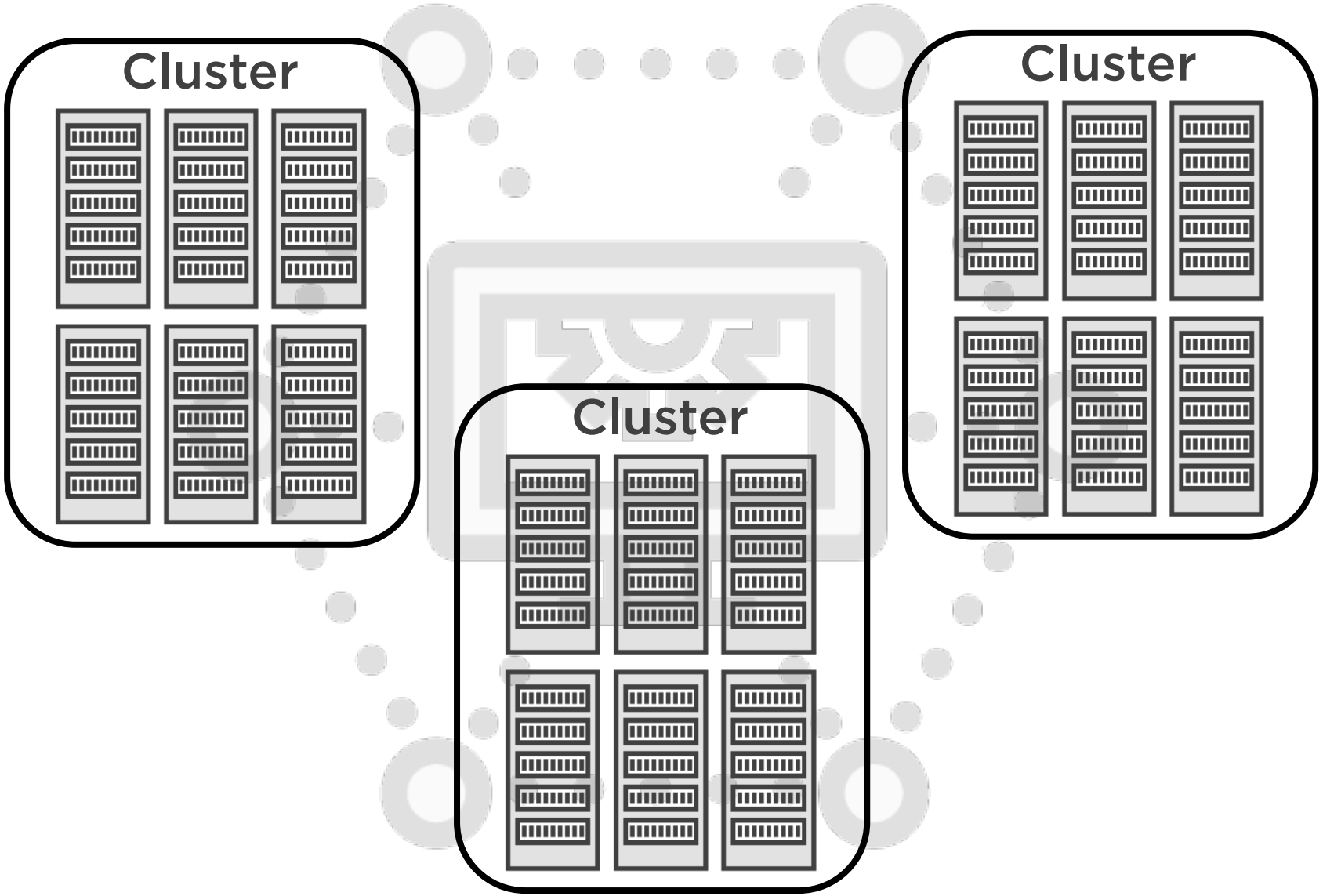
Kafka's Failure Manager: ZooKeeper

Apache ZooKeeper



Apache ZooKeeper

Apache ZooKeeper

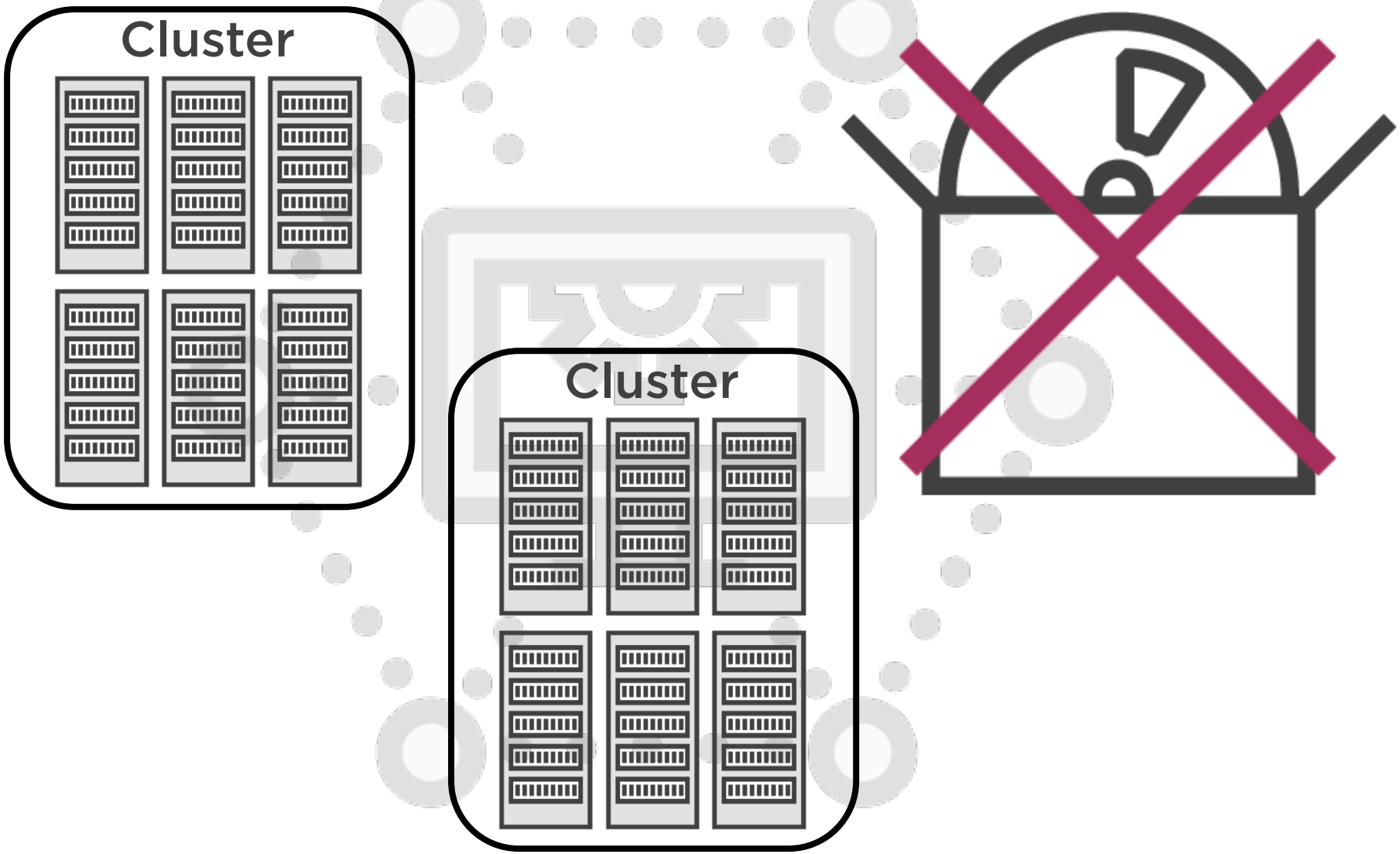


Apache ZooKeeper

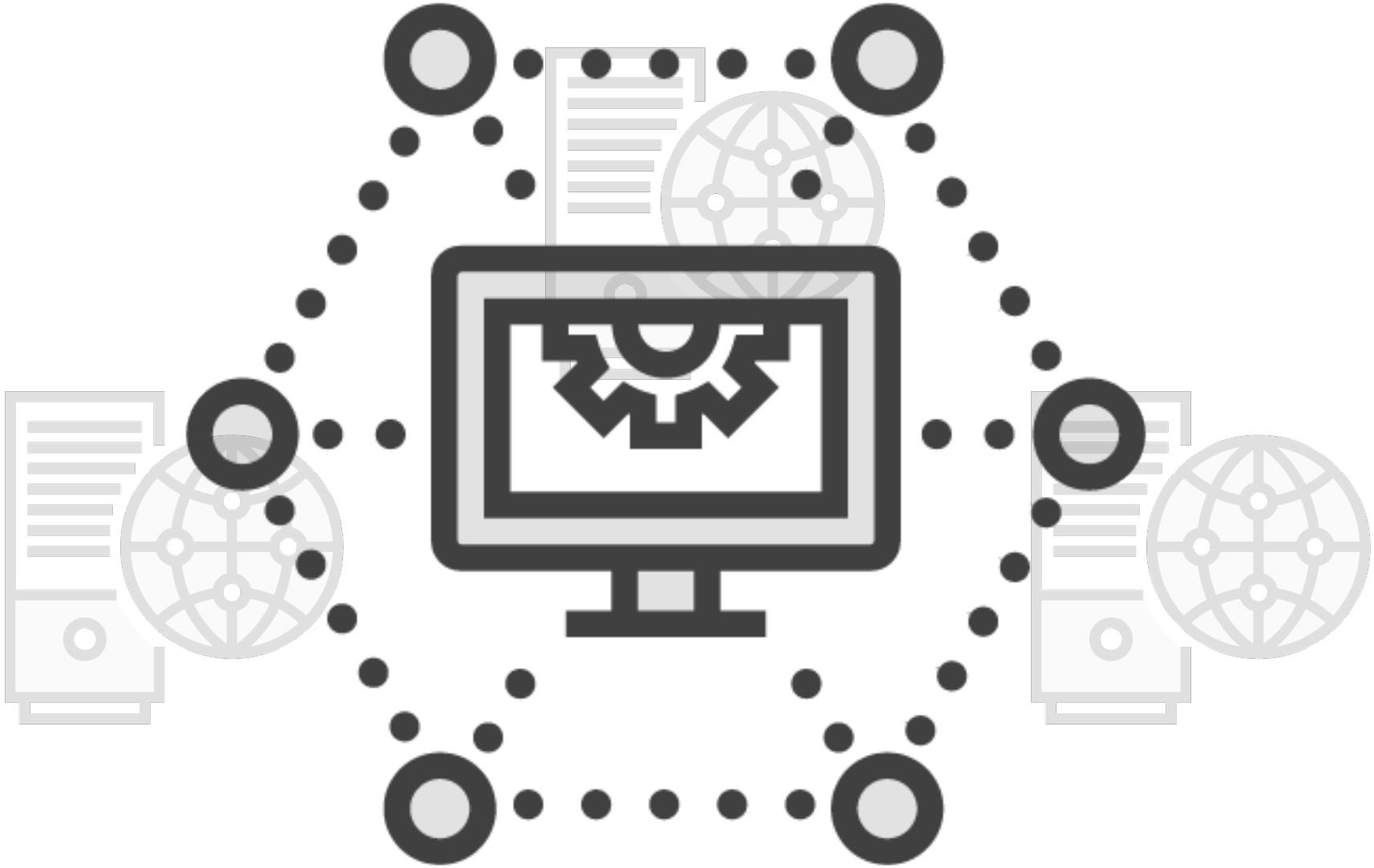
zookeeper.connect = host:port/KafkaCluster1

chroot

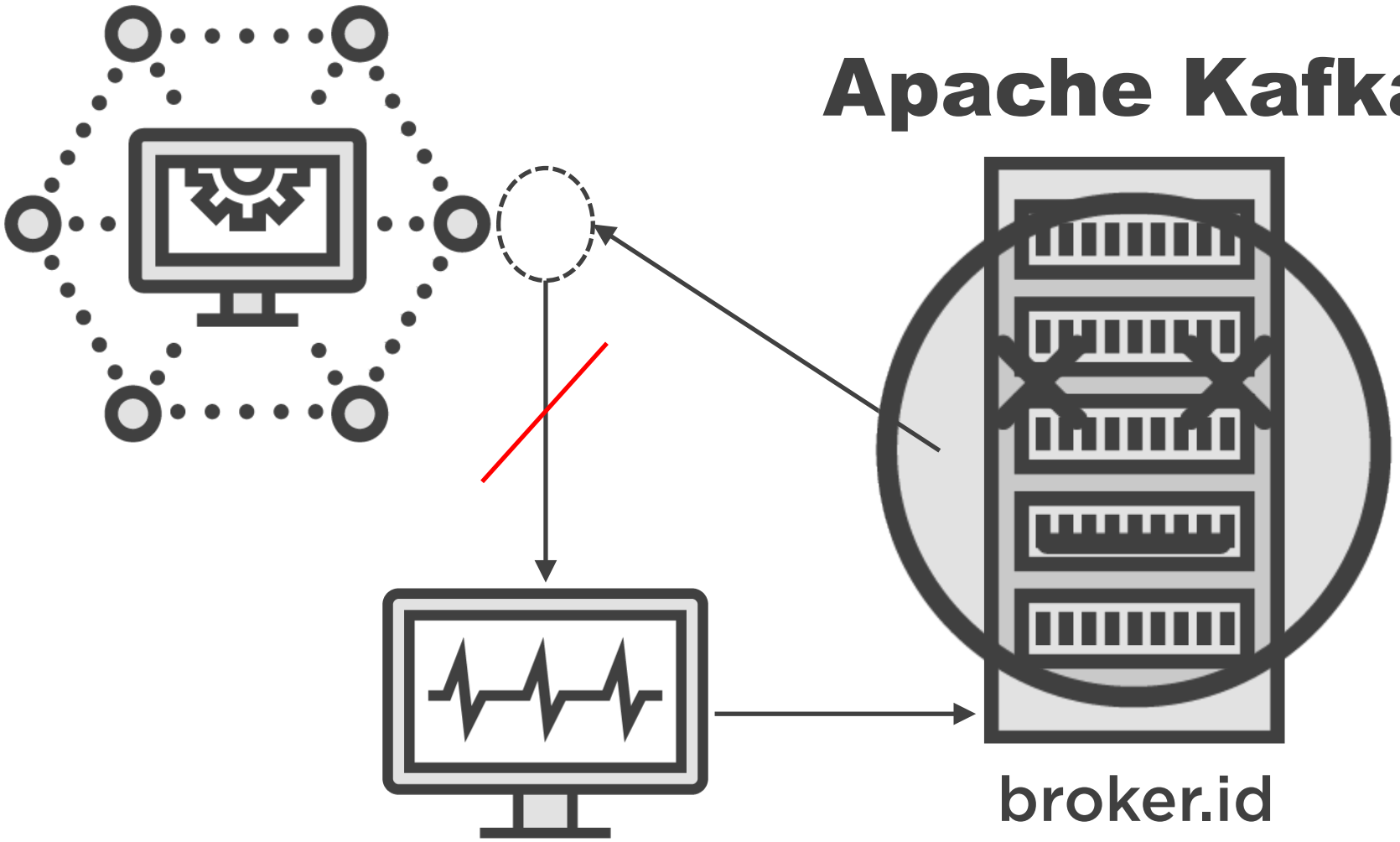
Apache ZooKeeper



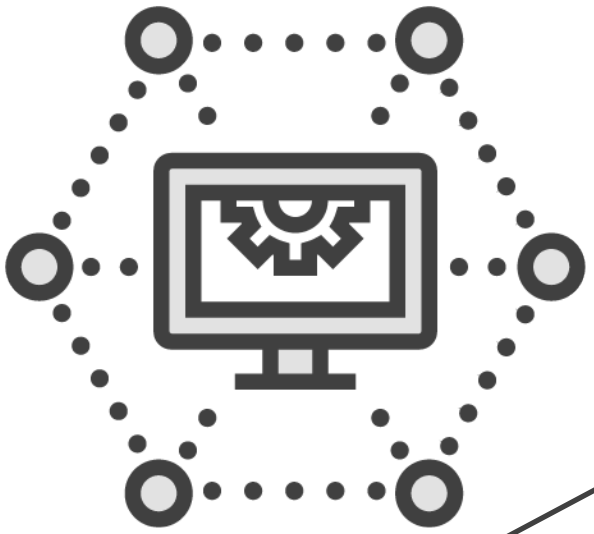
Apache ZooKeeper



Apache ZooKeeper

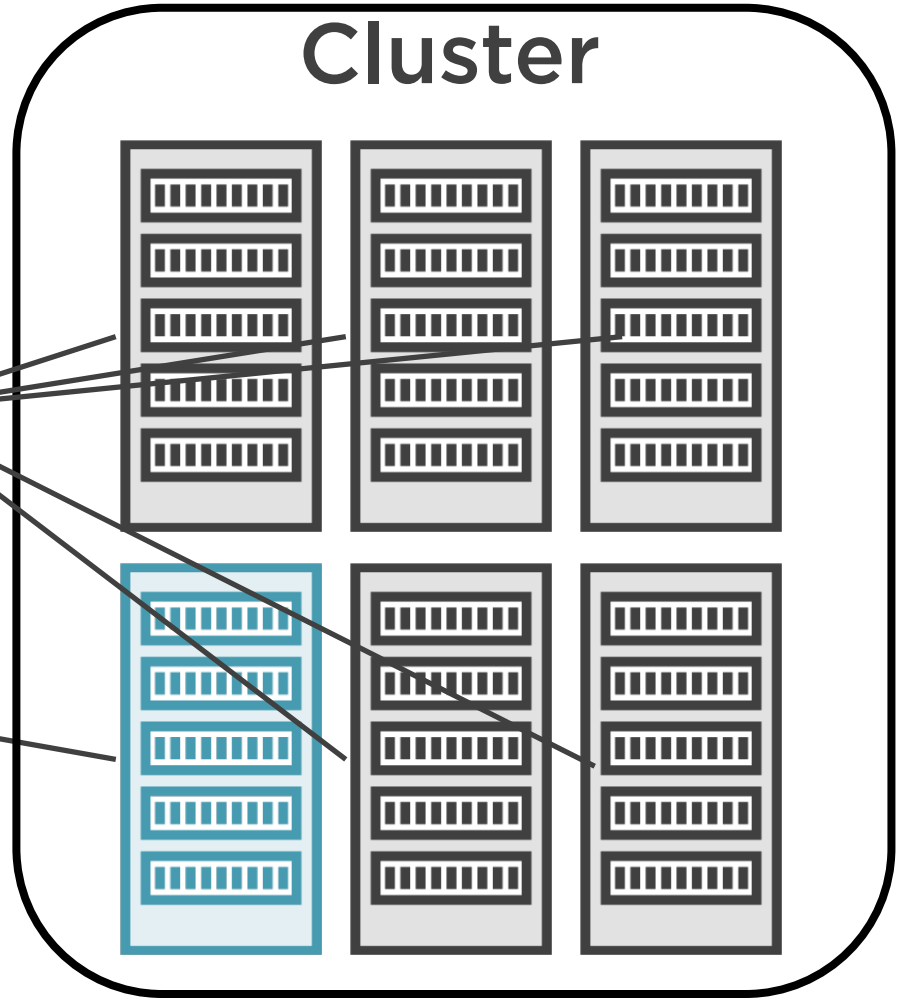


Apache ZooKeeper



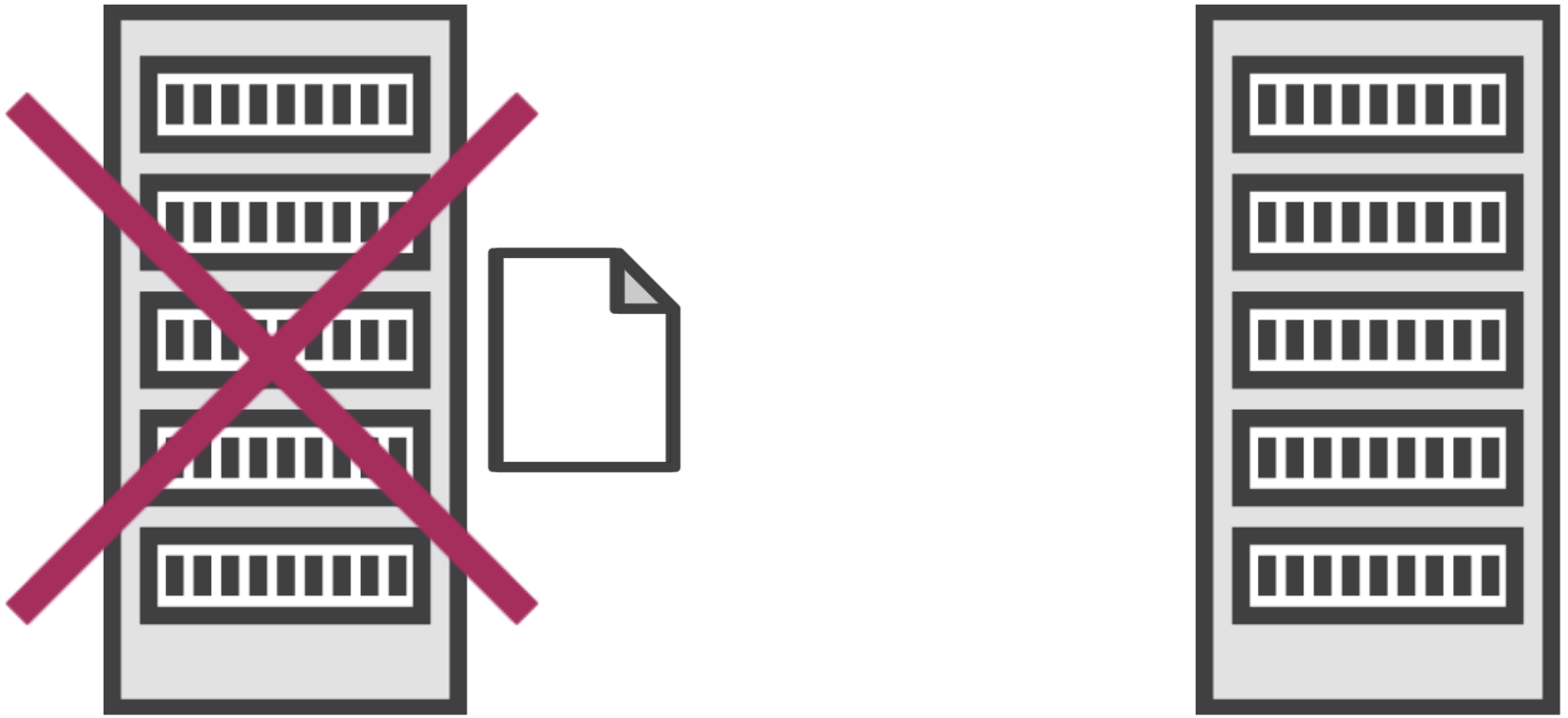
/controller

Apache Kafka

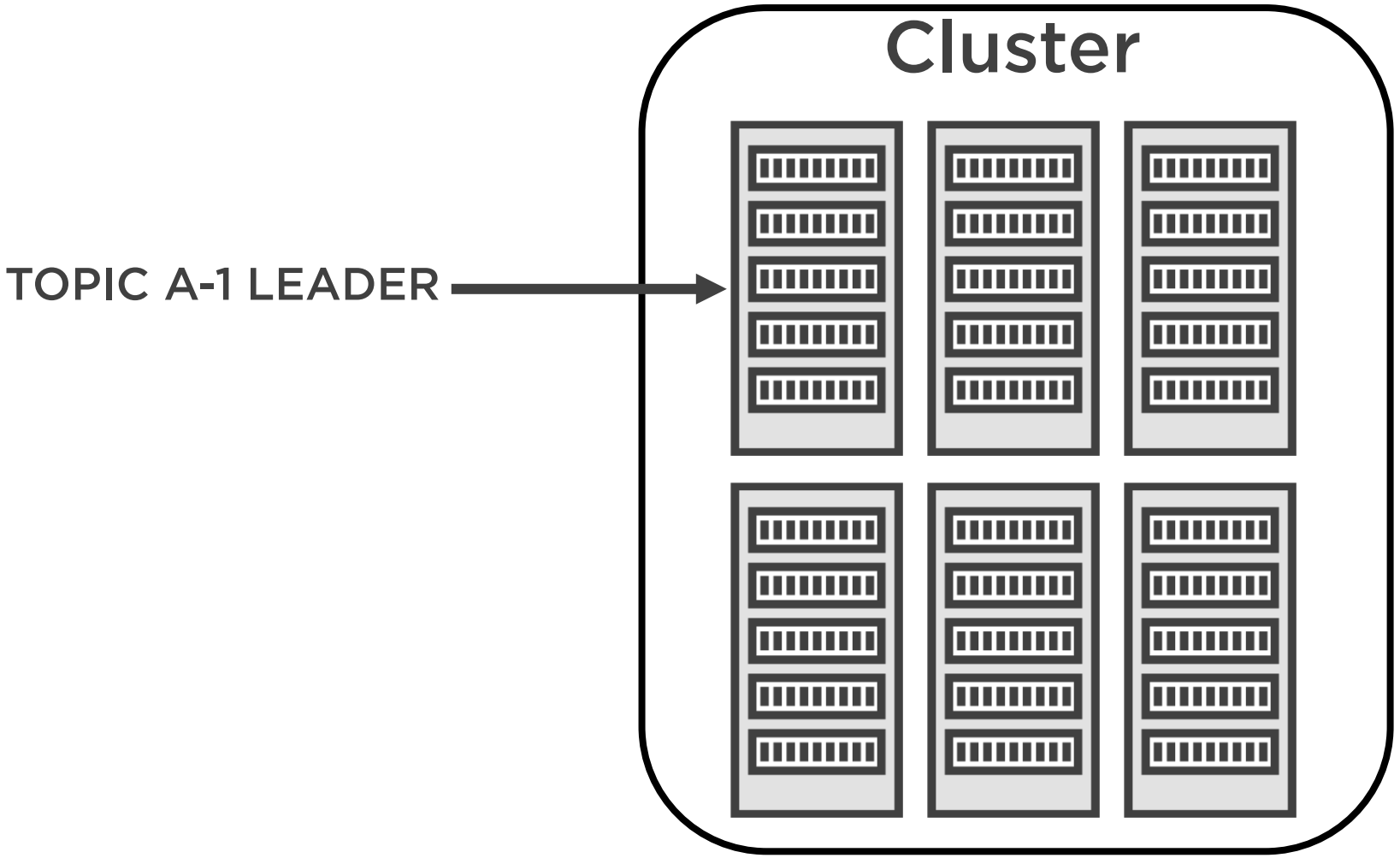


Keeping the data alive!
Data Fault Tolerance

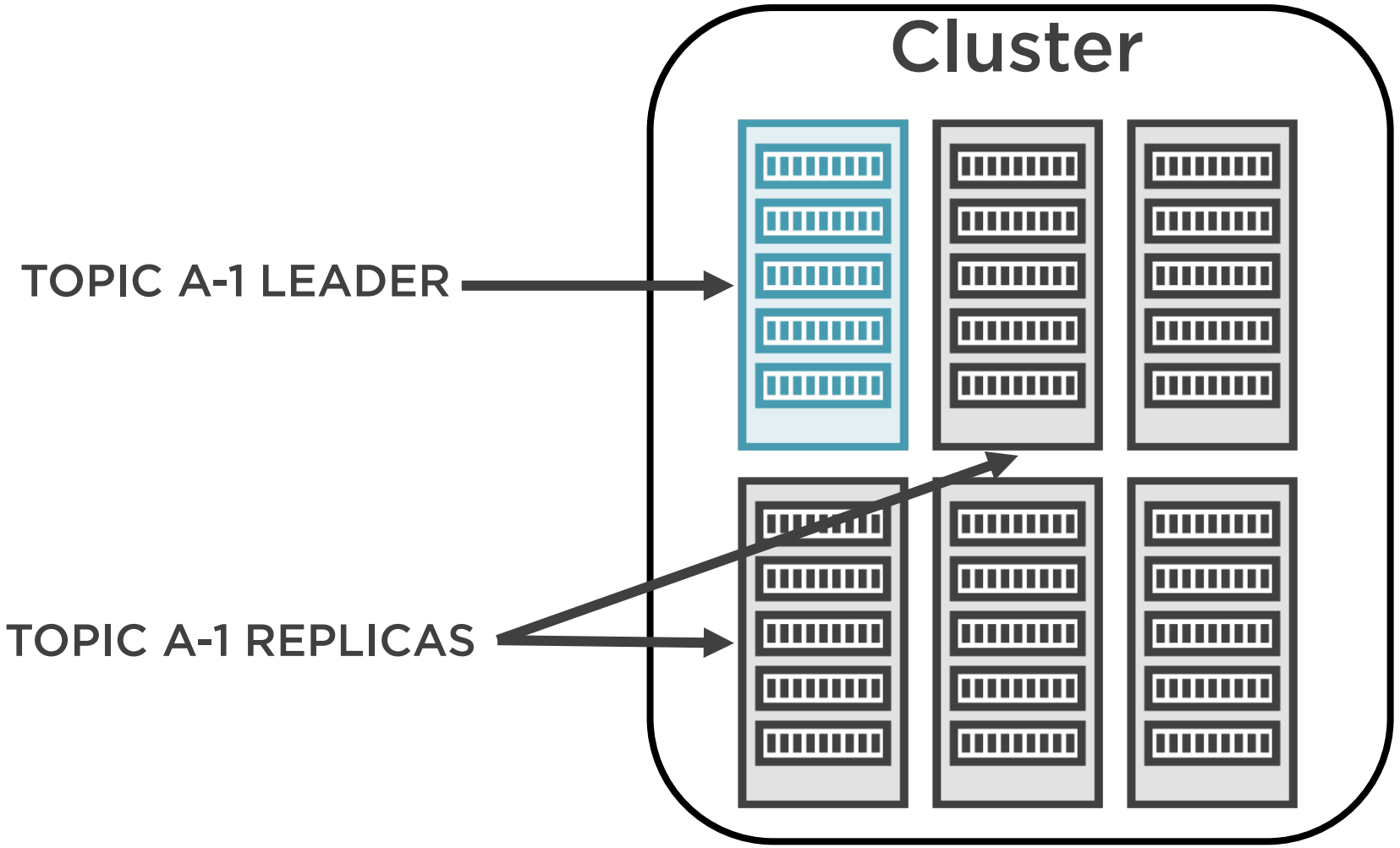
Replication



Replication

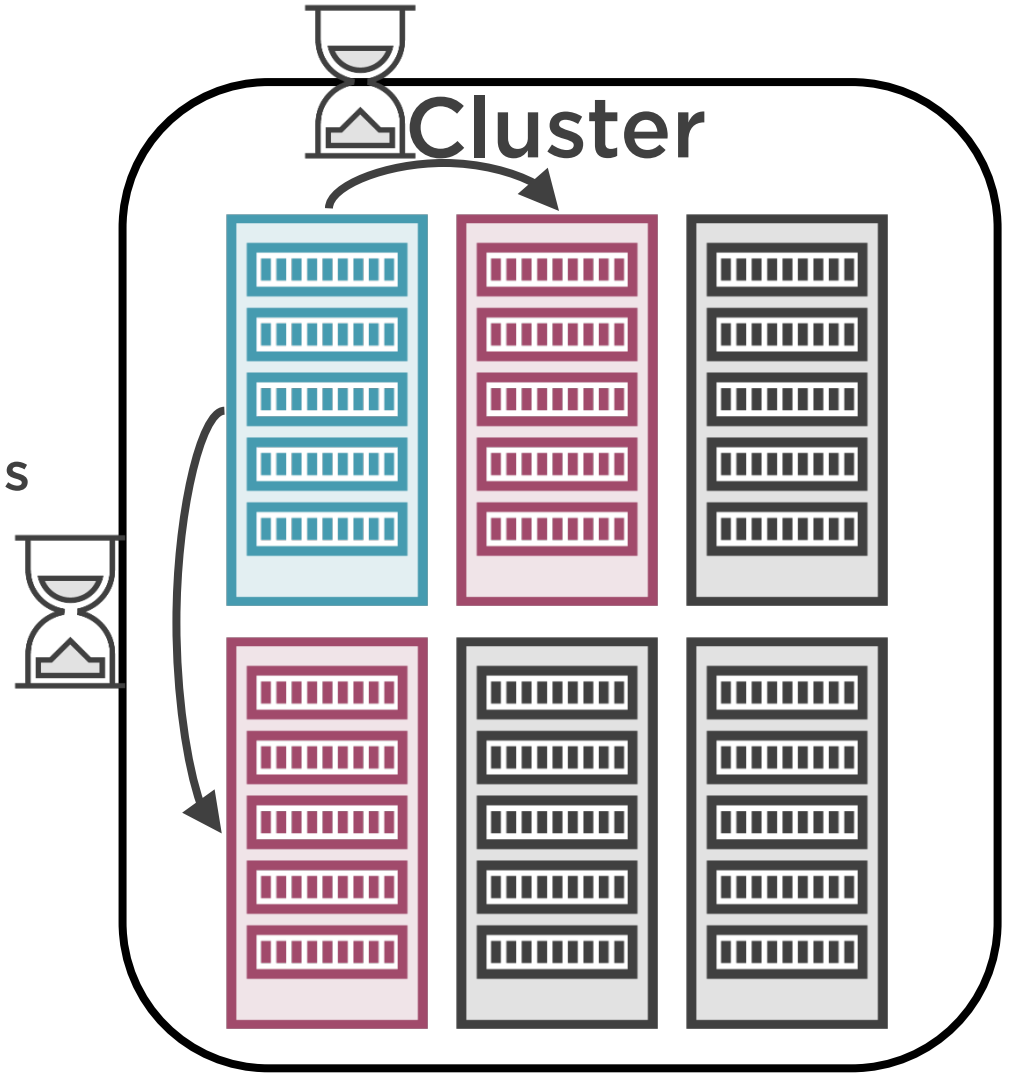


Replication

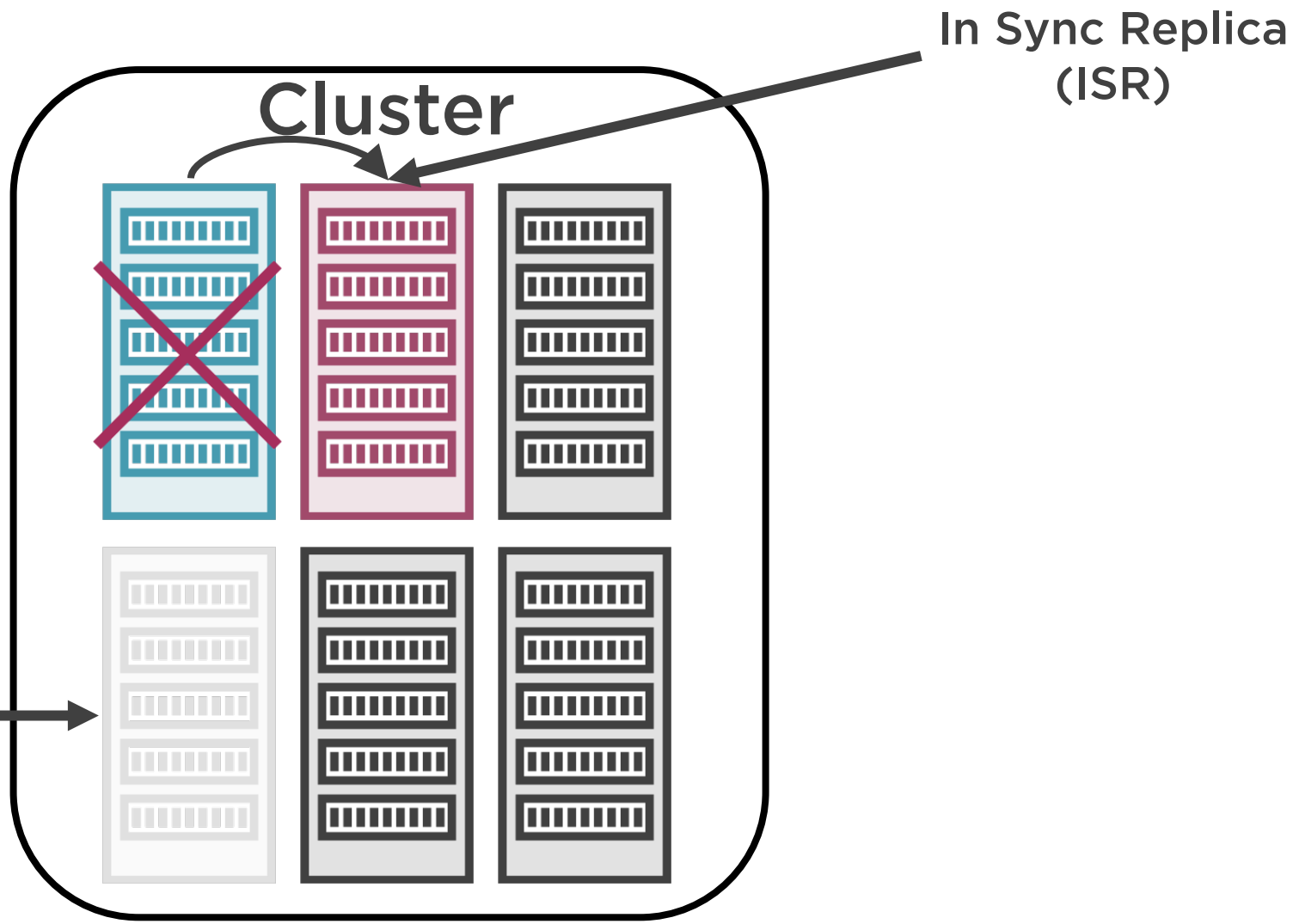


Replication

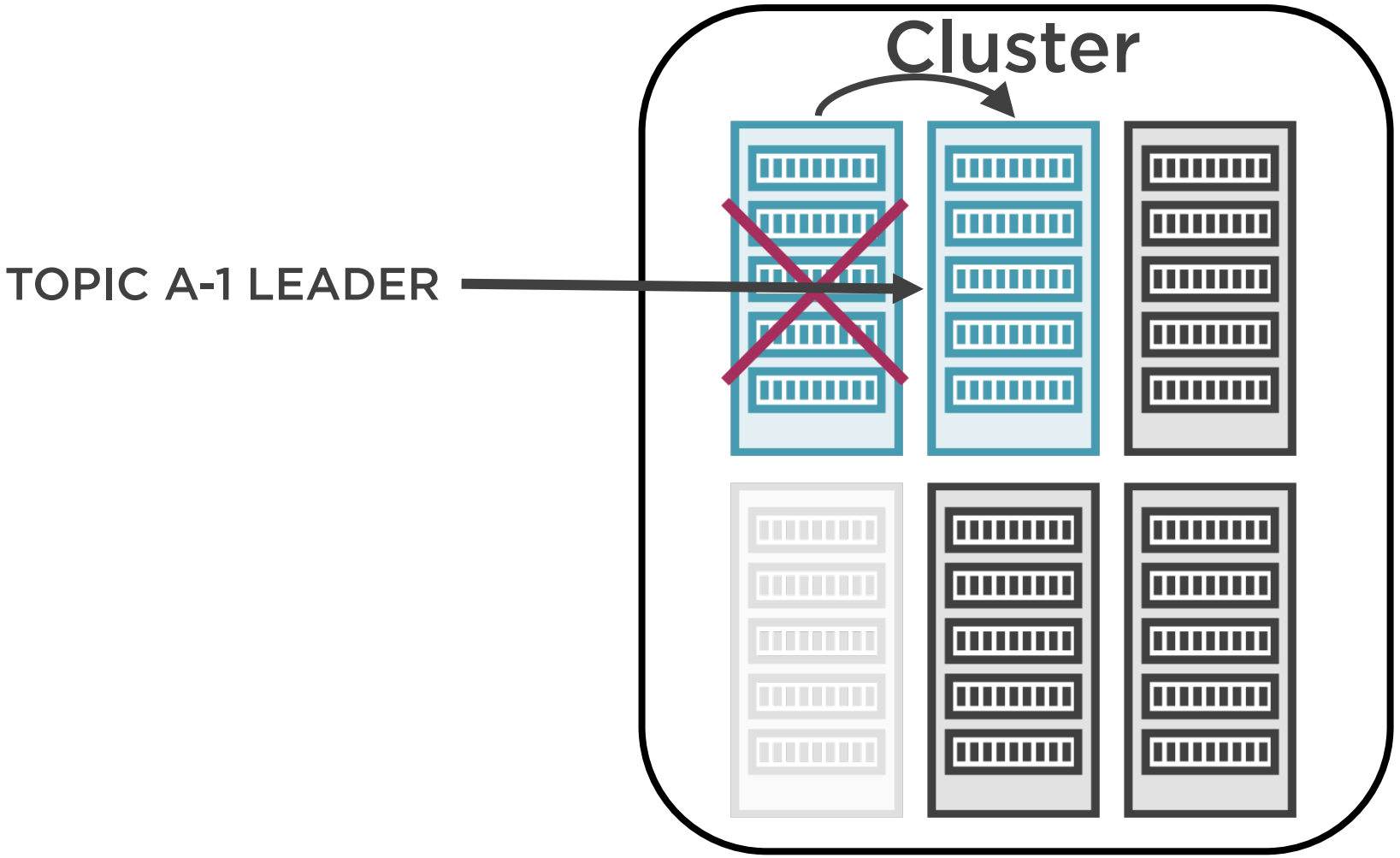
replica.lag.time.max.ms
10000



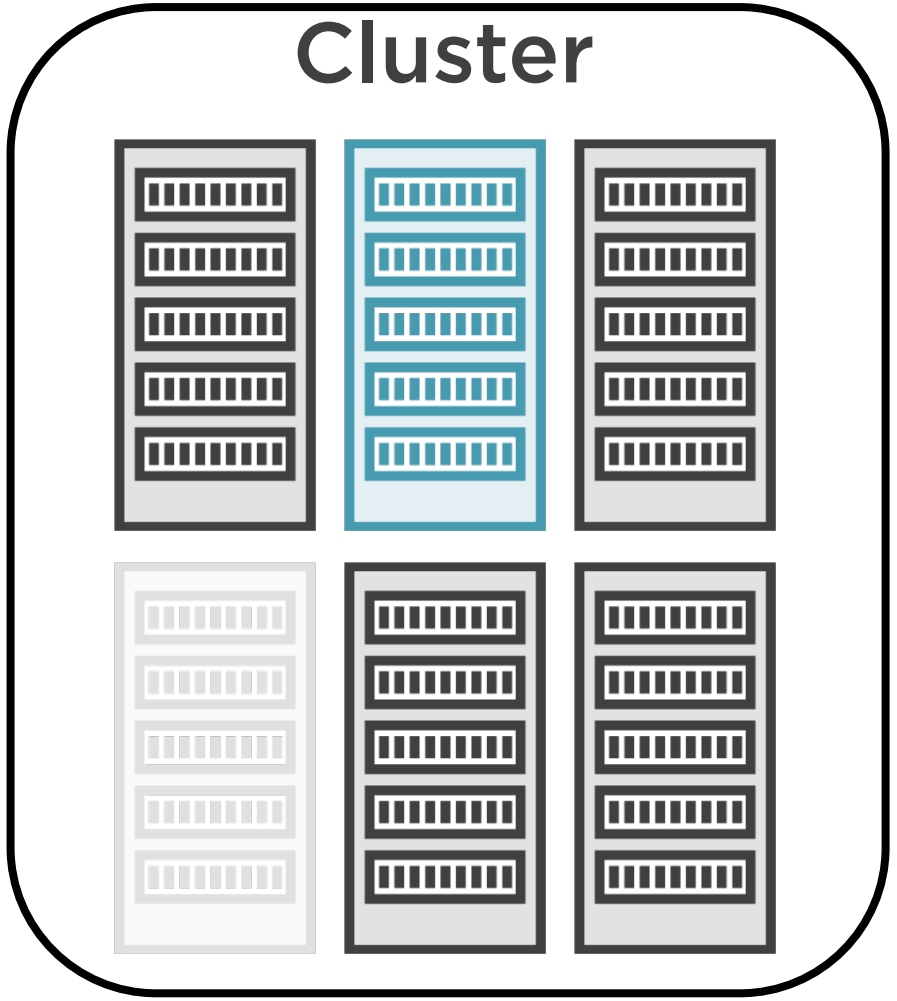
Replication



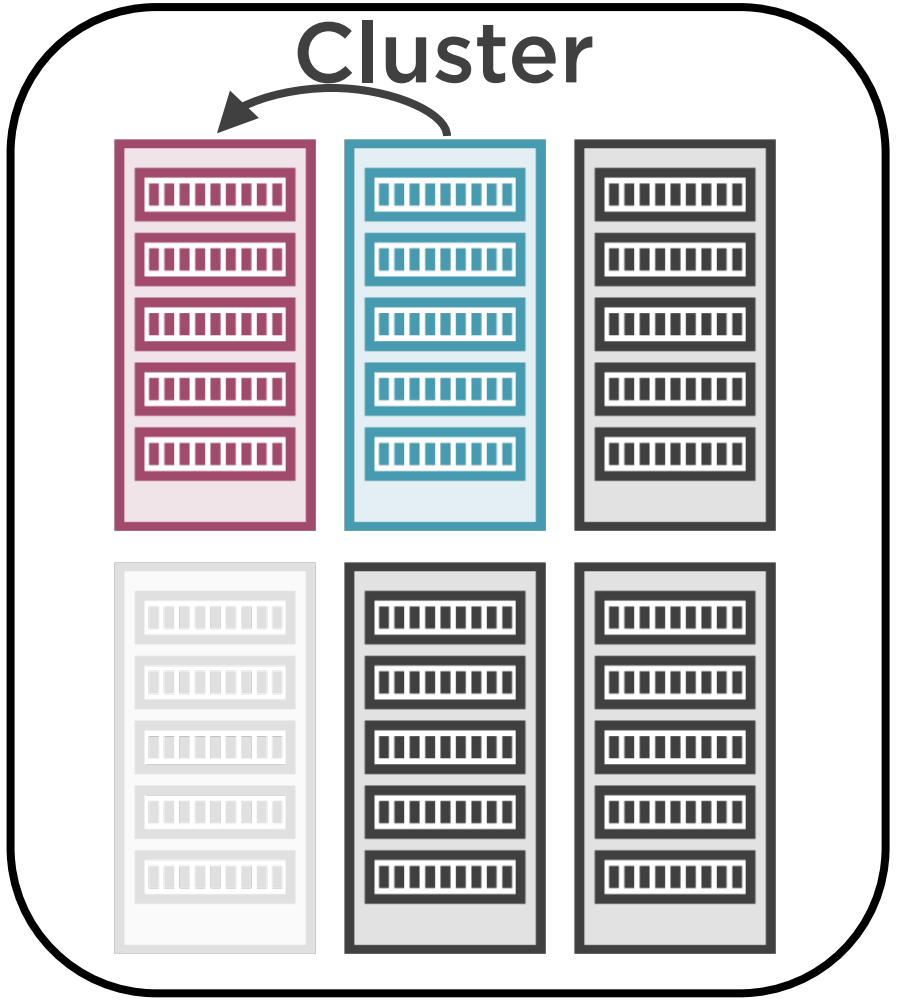
Replication



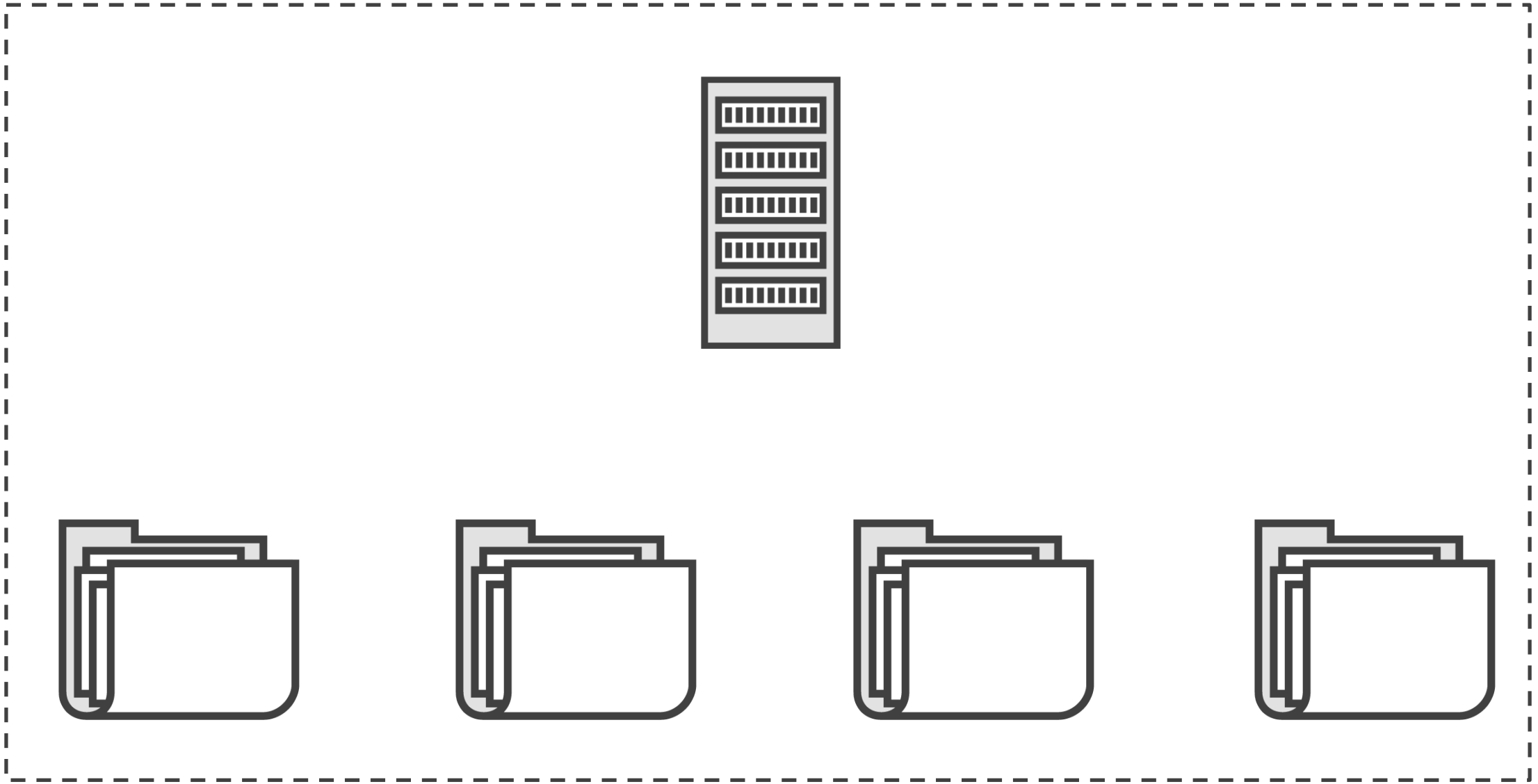
Replication

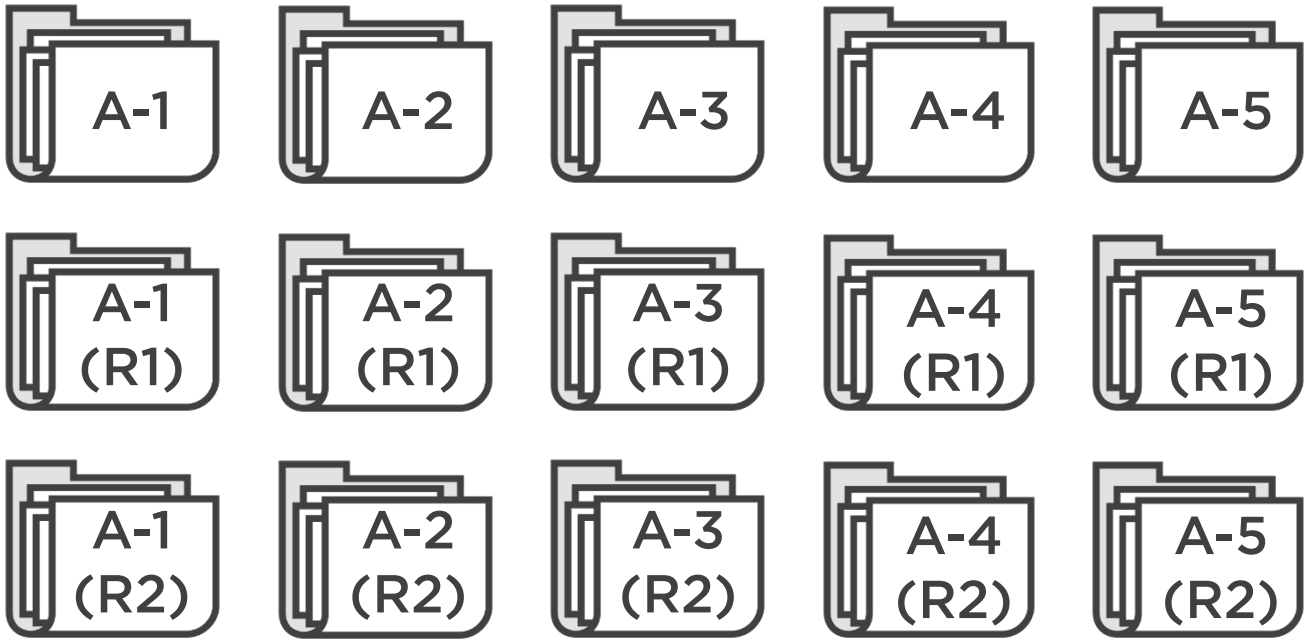


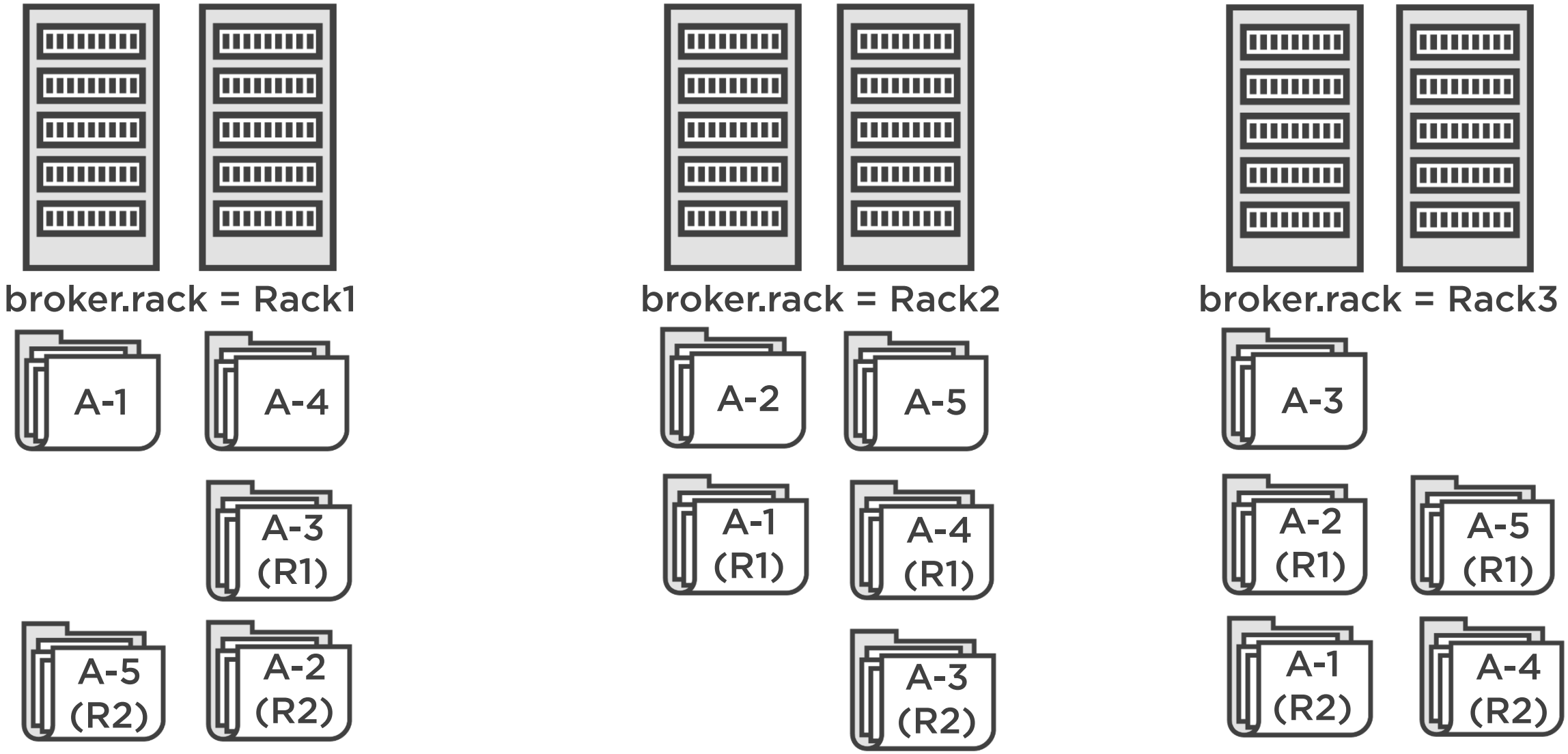
Replication



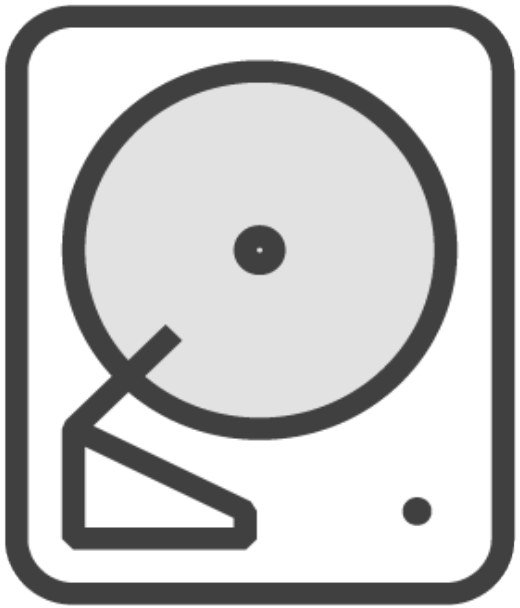
Data Management



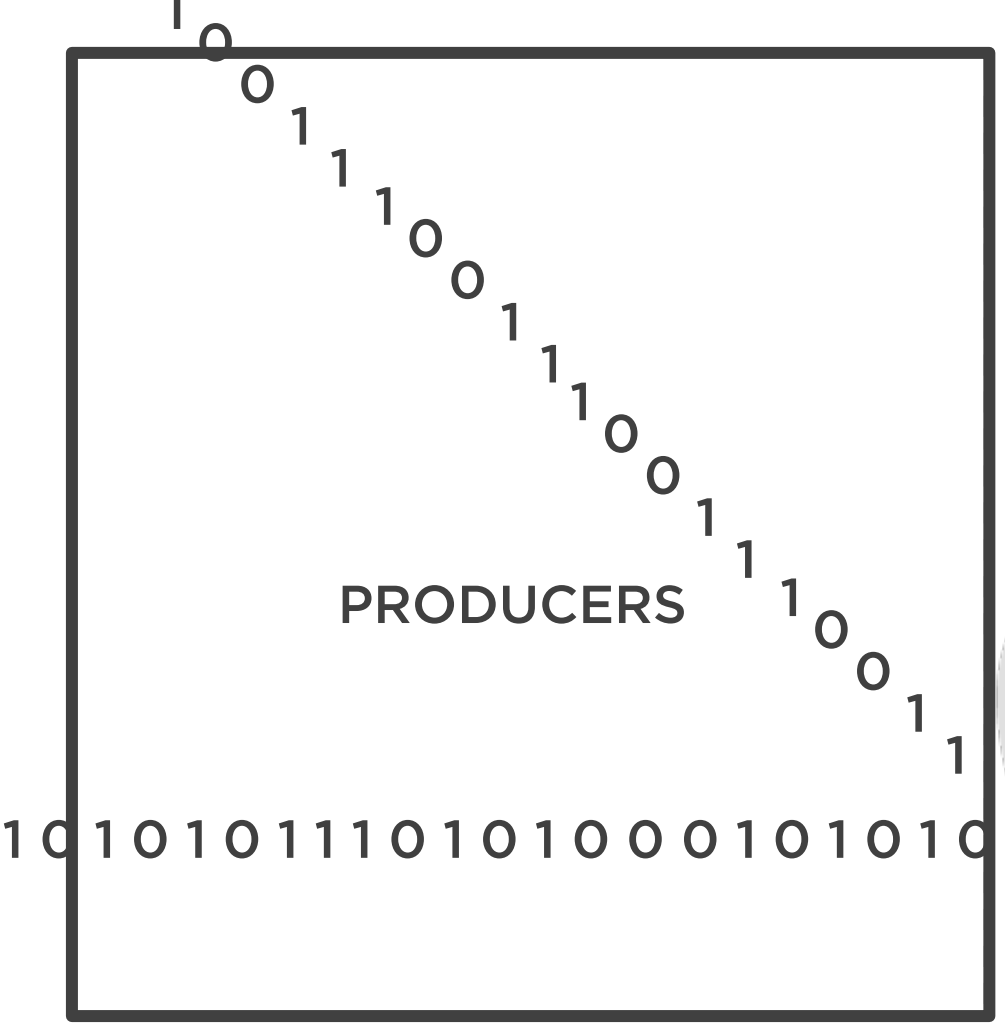

A-1
A-2
A-3
A-4
A-5
A-1 (R1)
A-2 (R1)
A-3 (R1)
A-4 (R1)
A-5 (R1)
A-1 (R2)
A-2 (R2)
A-3 (R2)
A-4 (R2)
A-5 (R2)

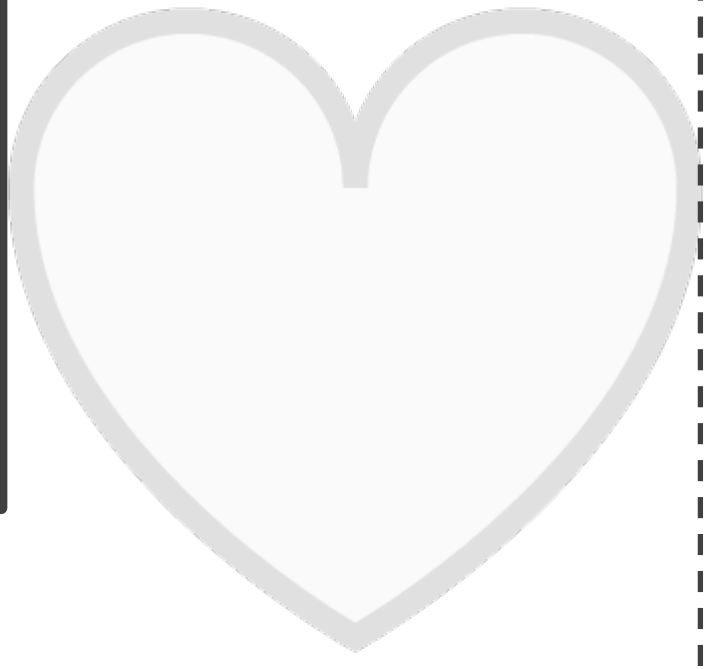

broker.rack = Rack1
broker.rack = Rack2
broker.rack = Rack3
A-1
A-4
A-3 (R1)
A-5 (R2)
A-2 (R2)
A-2
A-5
A-1 (R1)
A-4 (R1)
A-3 (R2)
A-3
A-2 (R1)
A-5 (R1)
A-1 (R2)
A-4 (R2)

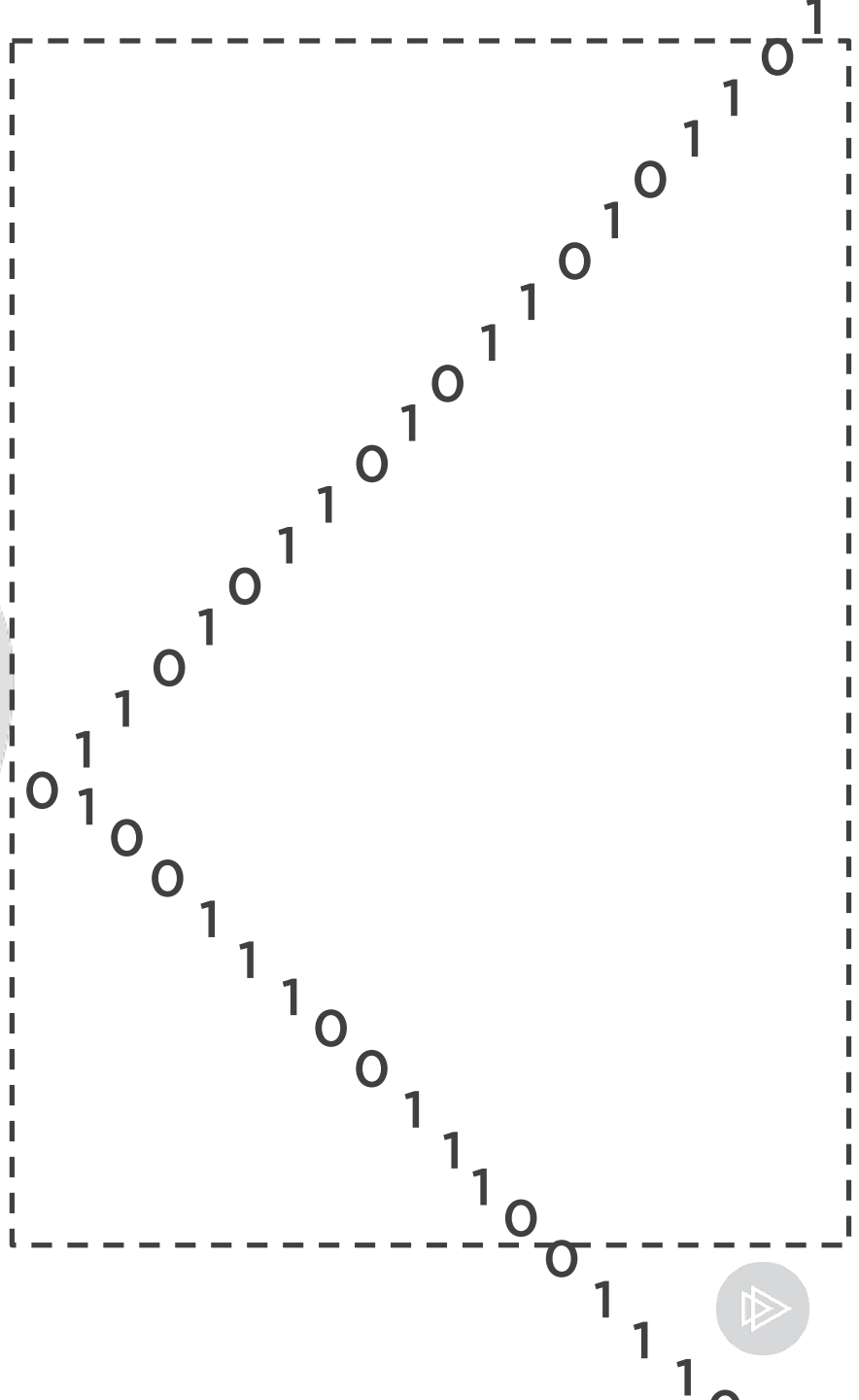
Data Retention



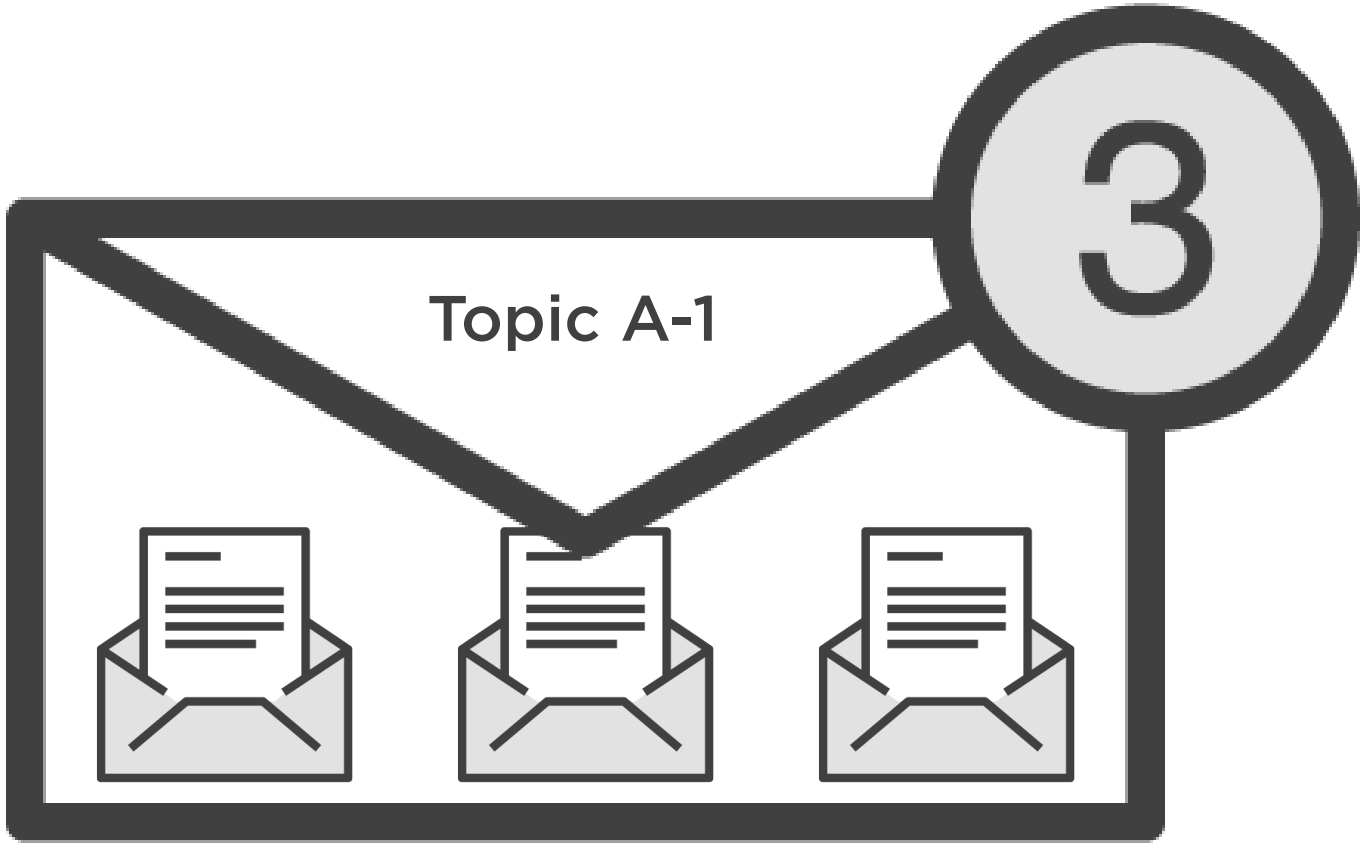
Kafka's Flow: Consumers & Producers


PRODUCERS
1 0 1 0 1 0 1 1 1 0 1 0 1 0 0 0 1 0 1 0 1 0

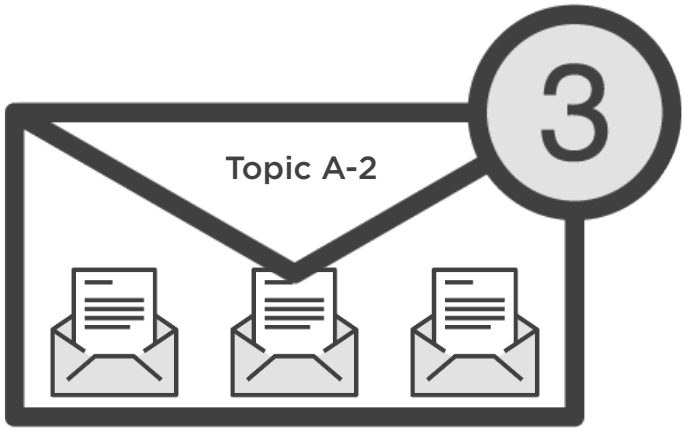


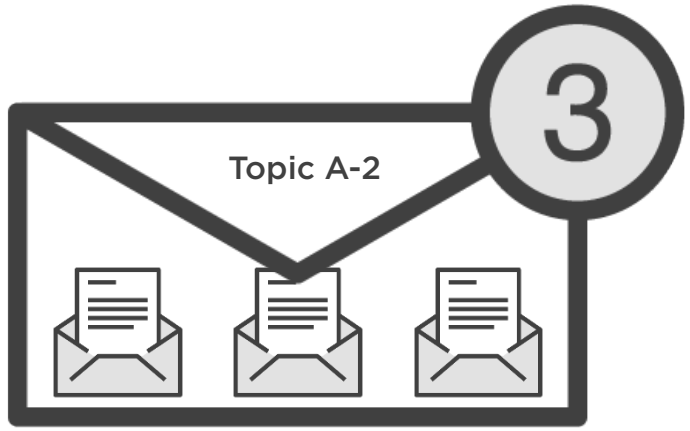


Producers



Producers

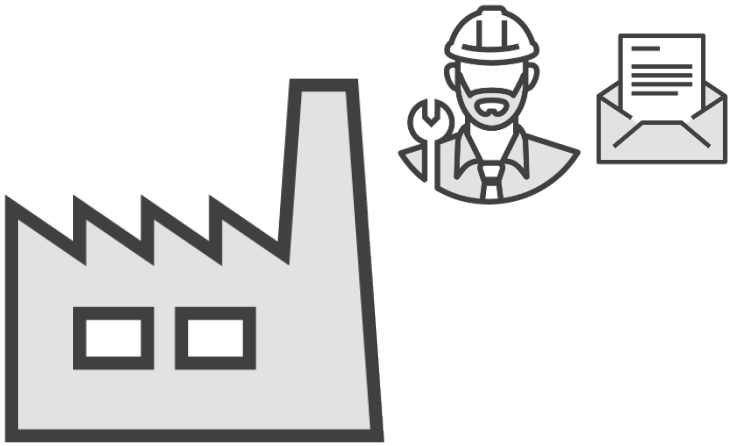




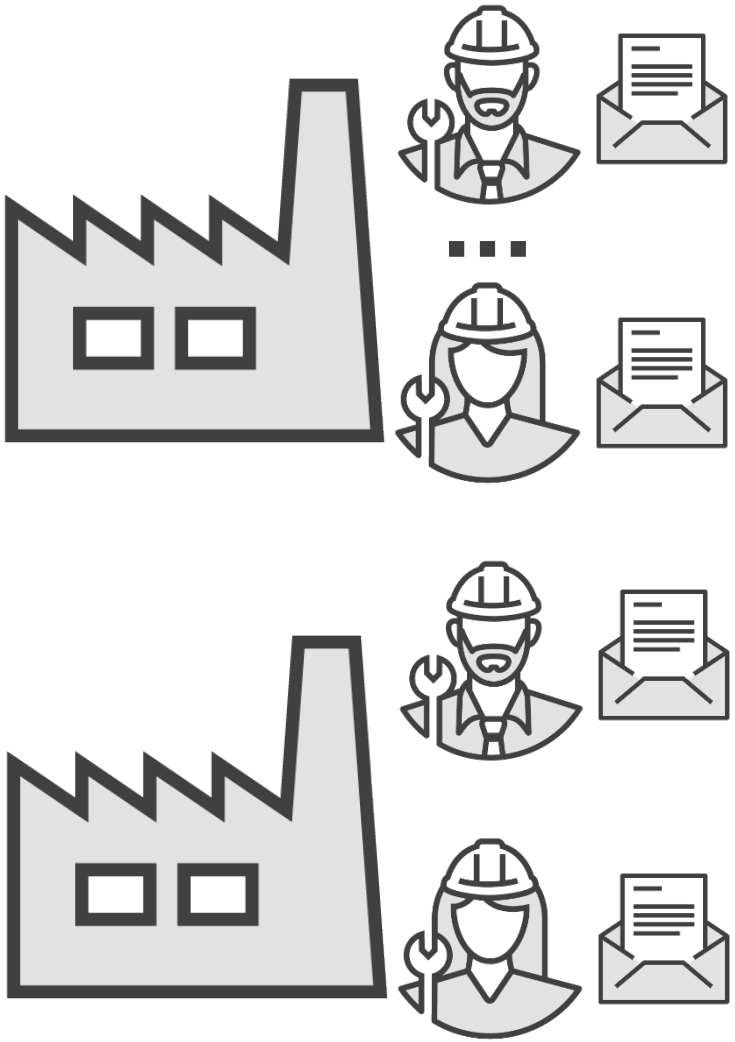
Producers

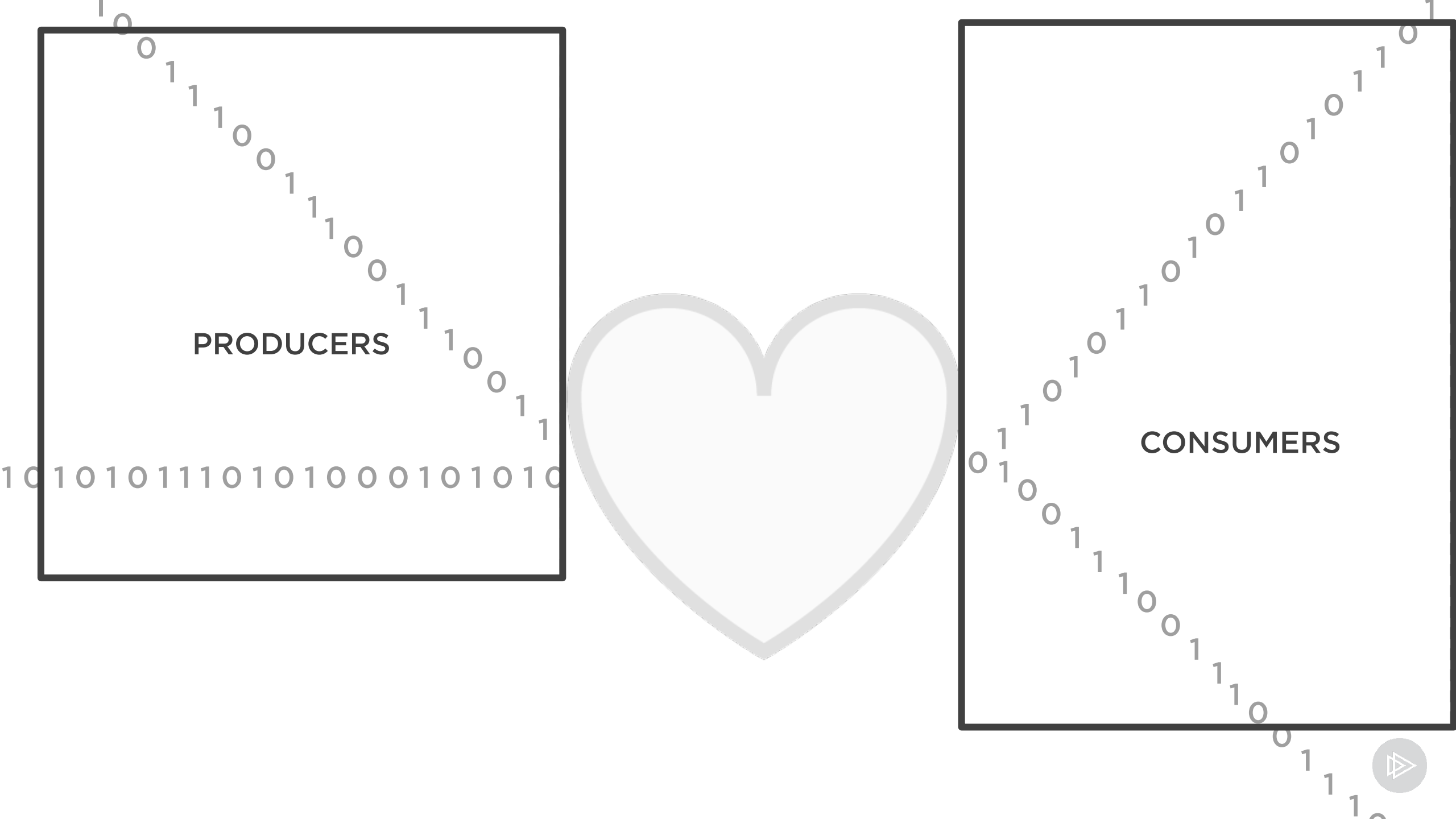
Topic A-?

Producers

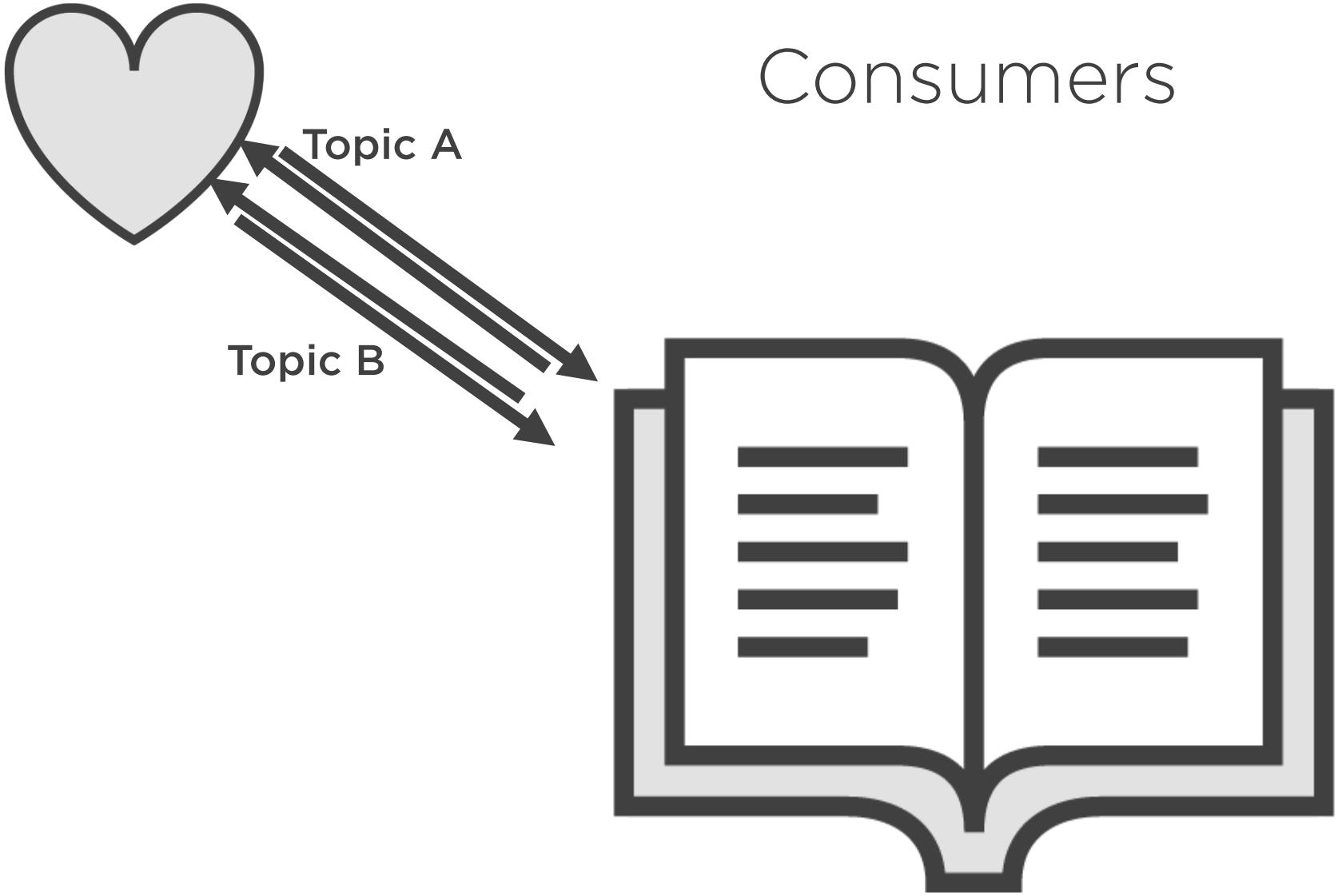


Producers

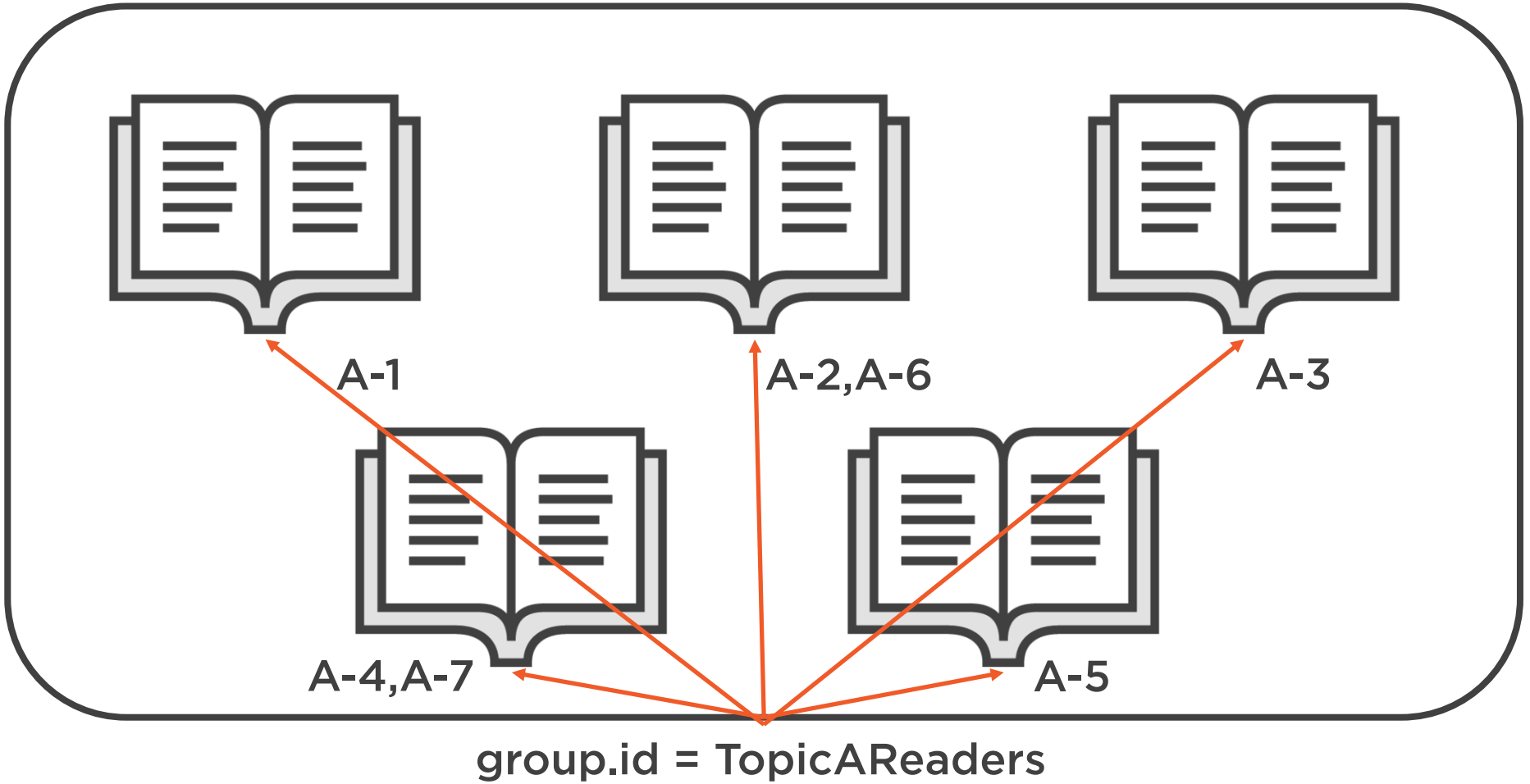



PRODUCERS
CONSUMERS

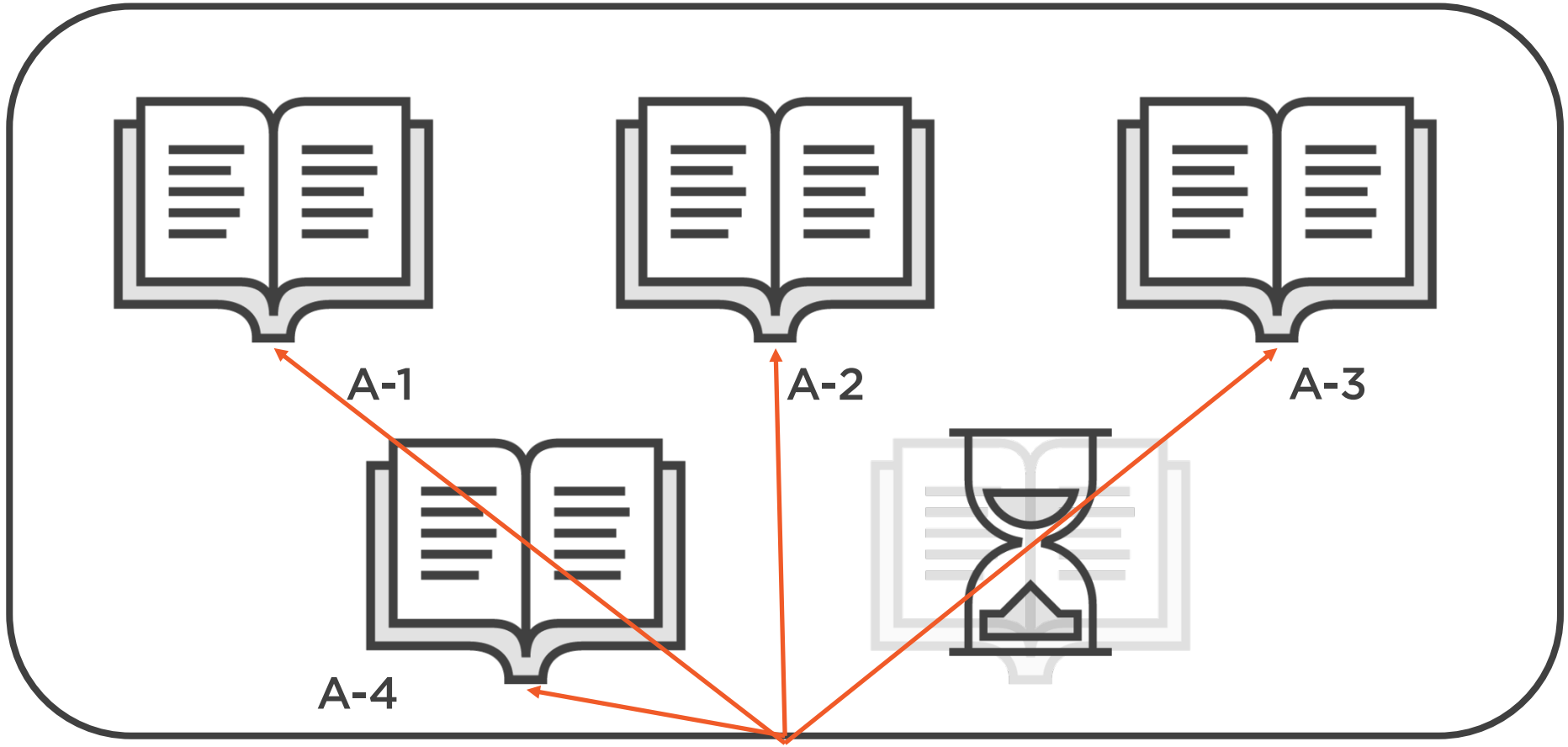
Consumers



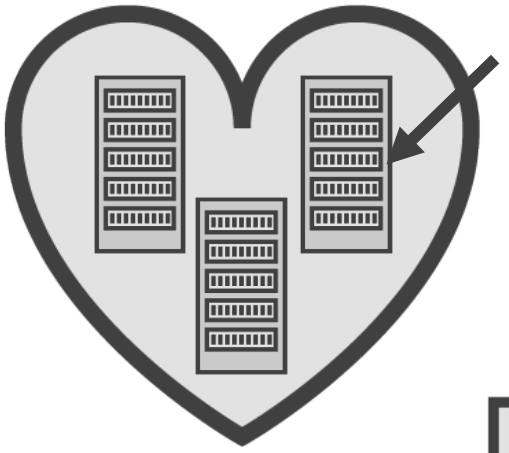
Consumer Groups



Consumer Groups



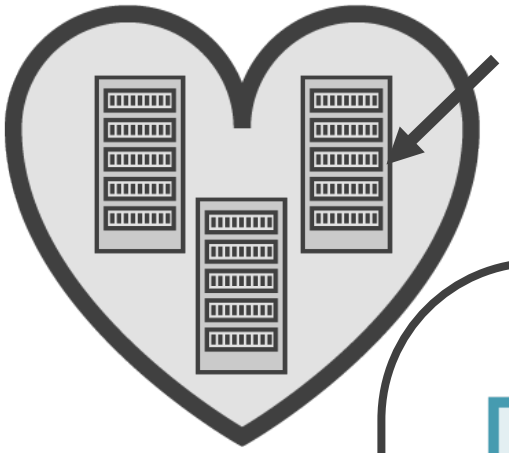
TopicA --partitions 4



Group Coordinator

Consumer Groups

group.id = TopicAReaders



Group Coordinator

Consumer Groups

LEADER

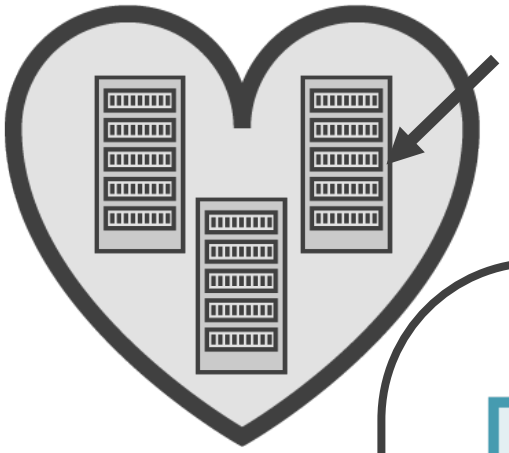
group.id = TopicAReaders
A-1

group.id = TopicAReaders
A-2

group.id = TopicAReaders
A-3

group.id = TopicAReaders
A-4

group.id = TopicAReaders
A-5



Group Coordinator

Consumer Groups

LEADER

group.id = TopicAReaders

A-1

group.id = TopicAReaders

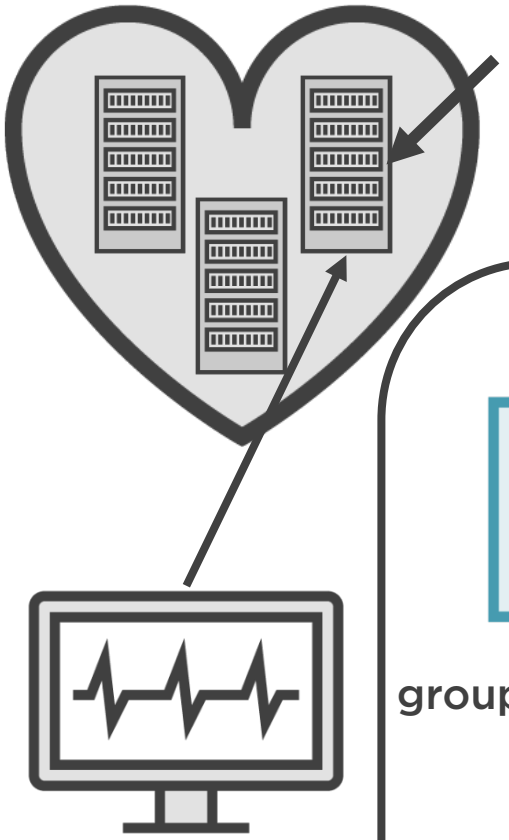
A-2

group.id = TopicAReaders

A-3

group.id = TopicAReaders

A-4



Group Coordinator

Consumer Groups

LEADER

group.id = TopicAReaders

A-1, A-5

group.id = TopicAReaders

A-2

group.id = TopicAReaders

A-3

group.id = TopicAReaders

A-4

Resources

- Apache Kafka Goes 1.0
  - https://www.confluent.io/blog/apache-kafka-goes-1-0/
- Quora on ZooKeeper
  - https://bit.ly/2MmyTPN
- Why Avro For Kafka Data
  - https://www.confluent.io/blog/avro-kafka-data/

Summary

Messaging for the Modern Era

Kafka Core Components

- Broker
- ZooKeeper
- Producer
- Consumer